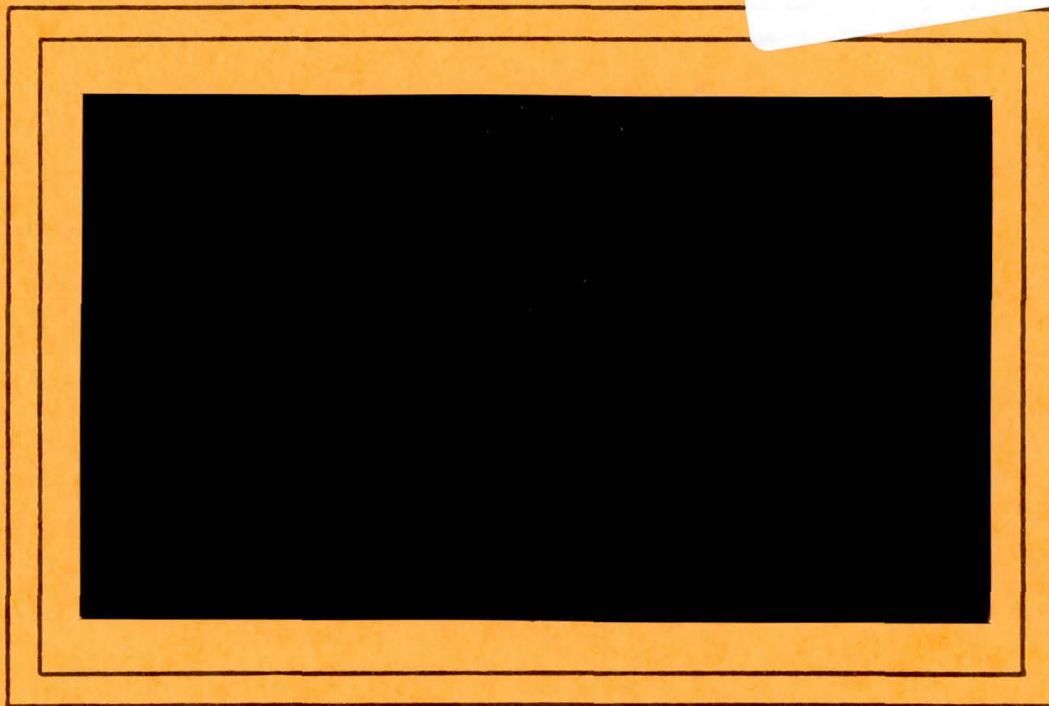


NASA CR-175295
19850025389



COMPUTER SCIENCE
TECHNICAL REPORT SERIES



UNIVERSITY OF MARYLAND
COLLEGE PARK, MARYLAND

20742

LIBRARY COPY

APR 8 1986

LANGLEY RESEARCH CENTER
LIBRARY, NASA
HAMPTON, VIRGINIA

COMPUTER SCIENCE: KEY TO A SPACE PROGRAM RENAISSANCE

Final Report of the 1981 NASA/ASEE Summer Study
On
The Use of Computer Science and Technology in NASA
Volume II : Appendices

Co-Directors

Raymond T. Yeh, University of Maryland
Paul B. Schneck, NASA/Goddard Space Flight Center

Edited by

Robert A. Freitas, Jr., Space Initiative
Patricia A. Carlson, Rose-Hulman Institute of Technology

University of Maryland
College Park, Maryland
January 15, 1982

105 33702#

Volume II: Appendices

- Appendix A. Final Presentation and Press Release
- Appendix B. Summary of 1981 Summer Study Technical Presentations
Author: Robert A. Freitas Jr.
- Appendix C. Survey of CAD/CAM Activities in NASA
Author: Barry Cooper, JPL
- Appendix D. The Need for Database Management Systems in NASA
Author: Sham S. Luthra
- Appendix E. A Vision of Future NASA Networks and Communications
Authors: Richard N. Draper, Temple H. Fay, Robert A. Freitas Jr.,
Richard H. Schippers, Richard S. Wallace
- Appendix F. An Executive Guide to the Computer Age
Author: William L. Benzon
- Appendix G. The Automated Work Environment: Opportunities for NASA
Author: Frederick B. Buoni
- Appendix H. NASA and Scientific Supercomputers
Author: Paul R. Healey
- Appendix I. Computer-Aided Design, Manufacturing, and Testing
Author: Roberta J. Ruth
- Appendix J. Expert Systems and Artificial Intelligence
Author: David M. Sandford
- Appendix K. Software Engineering and Technology
Author: Richard A. Helfman and Siew T. Koay

- Appendix L. Decision Support Systems
Author: Frederick B. Buoni
- Appendix M. Assessment of Computer and Communications Systems in the Private Sector
Author: Richard H. Schippers
- Appendix N. The NASA Manager Survey
Author: Richard G. Vance
- Appendix O. OPEN and ST: Two Future NASA Projects
Author: Riaz Hussain
- Appendix P. The High Cost of Computer Science Expertise
Authors: James C. Cleaveland and Richard N. Draper
- Appendix Q. An Inside View of CS&T in NASA: Problems and Potentials
Author: NASA Participants
- Appendix R. Report of the Goal-Setting Workshop

LIST OF FIGURES--APPENDICES

Figure D-1	Expected Data Volume	D-3
Figure E-1	A Vision of Future Networking In NASA	E-2
Figure E-2	Networks & Communications In NASA. A Plan for Phased Implementation thru Graceful Evolution.	E-3
Figure E-3	Phased Implementation	E-4
Figure E-4	Evolution of Tracking and Data Relay	E-8
Figure E-5	Evolution of Information Exchange Systems	E-13
Figure E-6	Responsive Scientific and Information Databases	E-15
Figure E-7	Evolution of MGMT. Information	E-17
Figure E-8	Responsive Technical Database	E-38
Figure E-9	Evolution of Technical Databases	E-39
Figure E-10	Evolution of Translation	E-40
Figure E-11	Automated Natural Language Translation Nodes	E-41

Figure E-12	Evolution of Hi-Bandwidth	E-42
Figure E-13	Future NASA Data/Information Accessibility Requirements	E-46
Figure E-14	Future NASA Turnaround Requirements for Data/Information Requests	E-49
Figure E-15	Future NASA Data/Information Accessibility Bandwidth Needs	E-50
Figure E-16	Information Factories	E-54
Figure E-17	Information Factories	E-56
Figure F-1	The Information Age In History	F-2
Figure F-2	The Computing Age	F-7
Figure F-3	The Temporal Dimension of Planning	F-9
Figure F-4	Computing and Productivity	F-11
Figure G-1	The NASA Workplace: Today and Tomorrow	G-3
Figure I-1	Computer Aided System Engineering	I-2
Figure I-2	Estimates of Yearly Costs In NASA Operational Activities With & Without Increased Automation	I-4
Figure I-3	CAD for Systems	I-15
Figure J-1	Sample Process Sequences for Expert Systems	J-5
Figure J-2	Benefits of Expert System Technology	J-7
Figure J-3	Application Areas of Some Existing Expert Systems	J-13
Figure J-4	General Areas and Modes of Use of Expert Systems	J-14
Figure J-5	Potential Applications of Expert Systems In NASA	J-15
Figure K-1	Software Life Cycle	K-3
Figure K-2	DOD Embedded Computer Software/Hardware	K-7
Figure K-3	Hardware-Software Cost Trends	K-8
Figure K-4	Most Errors In Large Software Systems are In Early Stages	K-9
Figure K-5	Software Validation: The Price of Procrastination	K-10
Figure K-6	Software Life-Cycle Cost Breakdown	K-11
Figure N-1	NASA Manager Survey Sample	N-2
Figure N-2	NASA Management: Years of Service	N-2
Figure N-3	NASA Management: Educational Levels	N-2
Figure N-4	NASA As A Research and Development Agency	N-5
Figure N-5	NASA As A Data Service Agency	N-5
Figure N-6	Evaluation of Computer Based Systems	N-5
Figure N-7	Experience With Computer Based Systems	N-7
Figure N-8	Computer Organization Support	N-7
Figure P-1	Supply of PhD Graduates	P-3

Figure P-2 Employment of Computer Science PhD Graduates

P-3

Figure P-3 Number of Faculty Holding PhD

P-3

Appendix A. Final Presentation and Press Release

COMPUTER SCIENCE: KEY TO THE SPACE PROGRAM RENAISSANCE

1981 NASA/ASEE Summer Study

University of Maryland/Goddard Space Flight Center

FINAL REPORT

August 14, 1981

NASA Headquarters Auditorium, Room 6104

9:30 A.M.

Opening Remarks,

Stan Sadin

for NASA:

Paul Schneck

Rodger Cliff

for the University of Maryland:

Raymond Yeh

Presentation of Final Report

9:30 A.M.

Introduction.	Frederick Buoni
Missions.	Richard Wallace
Systems.	Paul Healey
Information.	Temple (Buzz) Fay
Change.	William Benzon
Summary.	Richard Draper

Closing Remarks,

11:25 A.M.

for NASA:

Stan Sadin

Adjourn

11:30 A.M.

NASA NEWS

National Aeronautics and
Space Administration

Washington, D.C. 20546

AC 202 755-8370

For Release:

Ken Atchison

Headquarters, Washington, D.C.

(Phone: 202/755-2497)

IMMEDIATE

RELEASE NO: 81-119

SUMMER STUDY EXPLORES COMPUTERS AS KEY TO SPACE STUDIES

A dramatic extension of America's future space missions is possible if NASA vigorously pursues research and development of innovative computer technology now, according to participants in a recent workshop on "Computer Science: Key to a Space Program Renaissance."

To increase productivity both in space and on the ground and to make increasingly more sophisticated missions affordable, NASA is emphasizing the need for an extensive research and development commitment to computer science.

This jointly sponsored NASA/American Society of Engineering Education summer study was conducted at the University of Maryland's Donaldson-Brown Center in Port Deposit, Md.

August 14, 1981

Workshop participants studied complex, innovative systems technology which could influence the evolution of the nation's space program at every level, from office performance to the structure of new satellite missions and beyond.

The workshop team included a full-time interdisciplinary core group of 20 university researchers, assisted intermittently by more than 30 NASA engineers and project managers. The group has received technical and policy advice from Fortune 500 firms, high-technology aerospace and computer companies, various agencies of government outside of NASA and a variety of academic sources.

The 10-week workshop was completed Aug. 14, when an oral report was presented at NASA Headquarters in Washington, D.C., and reports were submitted for publication.

A key conclusion of the workshop is that adoption of an aggressive computer science research and technology program will:

- o Enable new mission capabilities such as autonomous spacecraft, reliability and self-repair, and low-bandwidth intelligent Earth sensing;
- o Lower manpower requirements, especially in the areas of Space Shuttle operations, by making fuller use of control center automation, technical support, and internal utilization of state-of-the-art computer techniques;
- o Reduce project costs via improved software verification, software engineering, enhanced scientist/engineer productivity, and increased managerial effectiveness; and
- o Significantly improve internal operations within NASA with electronic mail, managerial computer aids, an "automated bureaucracy" and uniform program operating plans.

The principal recommendation of the university faculty Study Group to NASA is to create a Headquarters computer science and technology program office. This office should, the group believes, encourage the establishment of

"critical size" computer science and technology enclaves at Agency centers and laboratories, and fund research and development at the cutting edge in such disparate fields as artificial intelligence and autonomous systems, advanced spacecraft computer systems, and database management systems for experimental data and scientific knowledge. The computer science program, if established, could generate long-range projections of computing needs, develop software engineering guidelines, adopt network policies and standards, and promote software exchange. Finally, the summer study participants recommended that specific responsibility be assigned to this office for monitoring the institutional effects of new technology, and for increasing the accessibility of organizational information.

The study further recommends that NASA:

- o Establish an automated work environment including executive and professional work stations. These stations should have access to decision support systems, word processing and electronic mail, effective local-area networks of distributed databases and data processing systems, agency-wide and external information bases, and computer applications packages.
- o Organize an agency-wide integrated data/information transport system to support scientific, engineering, managerial and administrative needs, possible eventually including general resource-sharing networks at the national level.
- o Coordinate and promote computer-aided systems engineering.
- o Develop aggressive recruiting and effective retention programs for computer science personnel, and improve professional interactions between NASA and the external computer science community.

The study was initially structured in teams which examined topics in three specific areas -- information systems, networks and communications, and computer science and technology. Additional conclusions and recommendations in each of these categories will appear in appendices to the final report,

which will be issued later this year.

Appendix B. Summary of 1981 Summer Study Technical Presentations8 June 1981 Introduction to the Summer Study

Stan Sadin from the Office of Aeronautics and Space Technology (OAST) presented a general charge to the Workshop to examine two main questions: (1) What can be done, using the benefits of computer science and technology, to make NASA more efficient and effective as the Agency is presently structured? and (2) How might greater effectiveness be achieved by restructuring the Agency itself?

Paul Schneck of Goddard Space Flight Center (GSFC) described the existing general organizational structure of NASA and suggested that the chief focus of the output of the Summer Study activity should be to produce a final product which can have major impact on the future organization and operations of NASA.

Rodger Cliff (GSFC) and Virgil Gligor (University of Maryland) discussed logistics, facilities, schedules and support which would be available to participants during the course of the summer.

9 June 1981 Libraries and Facilities Tours

Morning -- The Study Group toured the spacecraft mission control and worldwide communications facilities at GSFC, as well as the spacecraft assembly, centrifuge, and other facilities. Study participants were introduced to the GSFC Library, the staff, and the NASA RECON database system (which was demonstrated).

Afternoon -- The Group toured several libraries on the University of Maryland campus, available to all participants, and received library cards and maps.

10 June 1981 Creativity Workshop

Morning and Afternoon -- Under the guidance of Richard Horworth, Summer Study Fellows were introduced to the methodology of and possible barriers to

creative thinking, and participated in group activities designed to stimulate creativity using several distinct "hands-on" approaches.

11 June 1981 NASA Center Presentations

Ewald Heer presented an overview of activities at JPL, including mention of several possible future missions, with emphasis on the growing information-handling requirements of planetary and space sciences missions operated by JPL.

Susan J. Voigt and **John N. Shoosmith** described computer science and automation research programs currently underway at Langley. These include computer-aided design for aircraft (IPAD), ICASE, studies of fault-tolerant hardware and software systems, a small program in automation involving teleoperation/robotics and machine intelligence, and the potential for computer-controlled aircraft in the 1990's. Most of the large computers there were procured 15 years ago.

Max Engert briefly described the computer systems used in tracking and communications systems support of manned operations at Johnson Space Center.

Ralph Everett from Lewis Research Center described work in progress which includes massively centralized systems, the ESCORT computer system, and several applications programs involving computers such as maximally efficient heating of buildings.

Kenneth G. Stevens, Jr. described the state of computing facilities and programs at Ames Research Center. Ames has access to the ARPANET because of the ILLIAC-IV computer. Proposals include a state-of-the-art, high-speed mainframe computer architecture for high-volume computational applications, "dataflow" systems, and meta-control systems for managing otherwise incompatible operating systems.

Carl Delaune provided an overview of the Launch Processing System at Kennedy Space Center (KSC). The system requires on the order of 16 gigabytes mass-storage capability. Problem areas under review include commonality of

hardware and software systems, obsolescence, lack of a plan for phased replacement or facility upgrading, slow database accessibility, and a lack of rigorous measures of LPS performance.

Paul Schneck highlighted the peculiar disparity between the relatively large investment at GSFC in software development and maintenance versus the small number of computer scientists actually involved in the process.

12 June 1981 NASA Headquarters Presentations

Before the Headquarters speakers arrived, the Study participants produced a series of statements, questions, and phrases with which to solicit reactions from the Headquarters personnel. These issues of concern included:

- (1) Most challenging problem, nonpolitical?
- (2) Facilities (hardware, software, support personnel)?
- (3) Who are you? What do you do? Why are you here?
- (4) Obsession with successful projects?
- (5) How can we help you?
- (6) What is the relationship between HQ and the Centers?
- (7) What do you want/expect the role of your office to be in the future?
- (8) Paperwork crunch -- technical documentation, bureaucratic?
- (9) Financial constraints, buy vs. lease, GSA?
- (10) How would your office participate in a computer science program within NASA?

Michael Wiskerchen from the Office of Space Sciences (OSS) suggested that problems relating to the application of computer science and technology within NASA fall in the following categories: (1) Lack of coordinated planning for data processing, (2) technology and technology transfer, (3) education of NASA top- and middle-management, as well as scientific users, with respect to the benefits of computer science, and (4) the restrictive procurement environment.

William Raney from the Office of Space and Terrestrial Applications (OSTA) offered another list of CS&T failings within the Agency, including: (1) Need first-quality computer science personnel, (2) procurement difficulties, (3)

ability to manage and organize data of appropriate types, (4) need to improve capabilities in the area of automated materials processing using AI, and (5) need to examine critically the science underlying mission technologies. He also emphasized that hardware was not so critical a problem as software architecture, tracking required information through the system, and knowing what data to acquire and how to treat it for maximum utility for the ultimate user.

Donald Sova from the Office of Management Operations (Code N) raised as key issues the need for office automation, improved communications (e.g., networking, problem of rising costs of specialized systems), the burden of regulations (e.g., expensive, time-consuming), the need for new hardware to assist management in making prudent decisions, and improvements in software (e.g., specialized languages, standard high-level languages). Management goals for the future include updating the NASA ADPE inventory and reduction of procurement cycle time. Future technical goals involve software improvements, machine compatibility, integrated office environments, and networking systems.

Ronald Larsen from the Office of Aeronautics and Space Technology provided supplementary information regarding OAST and amplified remarks made earlier by Stan Sadin.

15 June 1981 Introduction to Computer Science

Maurice Wilkes discussed ring networks and Ethernet systems, including their general characteristics and modes of operation, data traffic rates, and the various "servers" which may be appended to each net.

16 June 1981 Database Systems/Software Engineering

Edward Sibley discussed issues in database management systems including query language options, database backup and security, the problem of deadlock and distributed data, and criteria for database systems evaluation. He then provided an overview of the basic architecture of a typical database management system, the goals ideally to be achieved (e.g., to make data available, to retain data over time, to control data, to reduce cost), and a

few future problems which should be considered -- such as the use of natural English as a query language.

Robert C. Tausworthe described the progress in software engineering at JPL, with directions for the future including critical near-term needs (e.g., technology transfer, better education and training), software cost and productivity improvement (e.g., increased manpower effectiveness), and various software development needs as identified in the Sagan Committee report.

E. David Callender spoke briefly and informally on the subject of information-intensive systems and complexity.

17 June 1981 Issues and Problems in Computer Science

Raymond Miller discussed the basic purposes of software modeling and described several theoretical models under current investigation which might be of use in formal software verification, network protocol analysis, programming analysis synchronization problems, hardware analysis, and general systems modeling such as Petri Nets, Dijkstra semaphores, schemata models and vector addition for parallel computation.

John E. Scull from JPL presented a historical survey of spacecraft computers, from early satellites which used watch mechanisms for timing to modern-day hybrid computer systems which employ arrays of IC chips laid out on a coated ceramic board with as many as thirteen layers of printed circuit planes (as on MJS 1977). A critical area for future advancement is fault-tolerant design.

18 June 1981 Software Engineering/Database Systems

Harlan Mills described software of the future from the perspective of structured programming (an orderly programming methodology intended to reduce debugging time and produce readily modifiable, easily-comprehensible, formally-verifiable "correct" software).

Frank McGarry of the Goddard Software Engineering Laboratory pointed out

that less than 1% of the funds expended on software development is used in R&D to produce better software development techniques.

Harry Sonnemann from Headquarters emphasized the need for NASA to join the ARPANET or some similar network facility to permit functional specialization and to share data, programs, and other resources among Centers.

Guy M. Lohman discussed Data Base Management Systems (DBMS) from the JPL perspective, raising a number of issues including data independence, multiple key queries, storage and access efficiency, integrity and security control, reliability, data retirement, and other specific problems relating to handling scientific data.

19 June 1981 Artificial Intelligence

Charles Rieger of the University of Maryland described the status of artificial intelligence in the area of natural language processing. Winograd's SHRDLU program and others were described in some detail, including concepts such as conceptual dependency, causal representations, and sentence parsing using "word experts." The ZMOB program was briefly described, a hardware system of 256 parallel processors, each of which can be programmed to run "word experts" in parallel to permit real-time sentence analysis.

Leonard Friedman presented models of human problem-solving activity, then described the Automated Problem Solver (APS) at JPL, various knowledge-based expert systems that have been developed, and several other issues such as non-monotonic logic and new approaches in dealing with uncertainty.

22 June 1981 Office Automation

Richard Harden presented material on information support structures for managers, a sample executive's equipment configuration, and keys to successful implementation. He then demonstrated two working executive information support systems which have the ability to help the manager control his time and information resources.

Harvey L. Jeane reported the development of the Automated Office Data Center (AODC Project) at JPL, a successful office automation system presently being deployed mostly at JPL but also at a few other Centers on a limited basis.

Michael Wiskerchen described a Headquarters viewpoint on the subject of office automation, including a recommendation to educate managerial people by starting with a simple, realizable pilot program.

23 June 1981 Human-Machine Interface

Ben Shneiderman discussed human factors engineering in connection with the design and use of programming languages, multi-user systems, text editing systems, and databases. A number of psychological issues were raised, including short-term memory load, closure of tasks, anxiety, and locus of control. A series of recommendations were presented, the most important of which was the re-emphasis of the role of computers as tools strictly subordinate to humankind (e.g., people as users rather than human-machine partnerships).

James F. Blinn of JPL described his recent efforts in computer animation and graphics, and presented several animated films displaying his work, including a sequence of DNA replication used in the TV series "Cosmos" and artistic conceptions of several recent planetary flyby missions to Jupiter and Saturn.

24 June 1981 Image Processing

Daniel Dudgeon discussed image modeling for segmentation and classification, iterative implementation of two-dimensional digital filters, and restoration of images using computer-processed reconstruction algorithms.

Mike Naraghi described work at the JPL Image Processing Laboratory.

Milton C. Trichel described JSC applications research areas pertaining to image processing, including (a) crop inventory and condition estimation, (b) forestry, range and land cover inventory and mapping, and (c) water

impoundment mapping.

John C. Lyon from GSFC outlined progress in imaging technology in the LANDSAT series of Earth-sensing satellites, giving various particulars and conditions of operation.

30 June 1981 NASA Communications

Jane Riddle demonstrated the NASA "RECON" database system to the group, provided the necessary access instructions to permit direct use by participants, then conducted a trial search on a sample topic of interest to the group.

Harold L. Hoff of OSTDS provided an overview of TDRSS (Tracking Data Relay Satellite System), in particular the process by which raw data are routed through the White Sands facility via domsats to JPL and GSFC. Specific performance characteristics and system capacities were briefly mentioned. Future projects beyond TDRSS now in planning include the TDAS (Tracking Data Acquisition System) and the ODSRS (Orbiting Deep Space Receiving Station).

15 July 1981 Kerrebrook Summer Study Discussion

Jack Kerrebrook, new Associate Administrator of the Office of Aeronautics and Space Technology, NASA Headquarters, spoke with the Summer Study participants. Dr. Kerrebrook requested that the meeting be conducted as an informal discussion rather than as a formal presentation.

Stanley Sadin (Code RS 5) introduced Dr. Kerrebrook to the group. Dr. Kerrebrook comes to NASA from MIT where he has been Department Chairman of Aeronautical Engineering for the past seven years.

In his initial remarks, Kerrebrook voiced a strong support of aeronautics and the extensive use of computers to support NASA's mission. He perceives that the Agency is behind in computer science. The evidence he offered was a lack of computer terminals at NASA Headquarters and Centers, as compared to an abundance of terminals at MIT and in many industries. He observed that at

MIT: (1) The students are integrating computers into their thought processes, whereas (2) the faculty was slow to understand that this had happened. About NASA, he noted (1) a failure to face the issue of how to provide sufficient computer capability, (2) the Centers still have traditional concepts related to centralized computer services as compared with modern distributed networks, and (3) NASA needs to lead in computer areas pertinent to its work. Another serious problem is that the aging NASA work force is obsolete with respect to new computer technology. NASA needs to improve in-house competence in computer technology with ultimate attention given to money, people and bureaucracy problems.

Much of Dr. Kerrebrook's discussion centered on the proposed Numeric Aerodynamic Simulator (NAS) and focused on the following points:

- (1) NAS is a good idea and extremely important to NASA's future.
- (2) NAS is a means for Ames Research Center to obtain the most advanced possible computer to solve fluid and structural mechanics problems.
- (3) Dr. Kerrebrook was concerned that NAS was evolving into a facility rather than a research and development effort. It is expected that the system will evolve and remain on the cutting edge of computing technology. The initial \$100 million cost is the first step in this direction.
- (4) The grandfather of NAS, Neil Lincoln of Control Data Corporation, was present during the discussion and raised serious concerns about the direction and escalating funding of NAS. Dr. Kerrebrook invited Neil to join him and the corresponding Burroughs corporation counterpart to meet at NASA Headquarters prior to any further decisions about NAS. Dr. Kerrebrook said that no decision on NAS is required until the Spring of 1982 and he solicited input from everyone prior to that time.

The guest speaker also raised the question of how to upgrade existing NASA Center computational capabilities. He cited the NASA Langley CYBER 203 as the only "Class 6" computer anywhere in the Agency. Plans exist to have a "Class

6" computer at each major Center: Ames (Cray 1 being installed), Lewis, Goddard, and an upgrade at Langley.

Dr. Kerrebrook stated that a satellite datalink will exist between NASA Centers, enabling wide-bandwidth communications to take place from site to site. He stated that the biggest increases in capability will come about through advances in information handling, distributed computation, and reliability (e.g., graceful degradation, fault tolerance, and analytical redundancy). He solicited ideas on NASA involvement in Air Traffic Control, in particular: (1) An integrated avionics system with significant computational capabilities in the aircraft, and (2) the technological contribution of NASA towards more autonomy in Air Traffic Control.

Dr. Kerrebrook concluded the session by retracing the NACA and NASA roles in aeronautics and suggested a corresponding thrust in support of the commercial satellite industry might be appropriate.

Appendix C. Survey of CAD/CAM Activities in NASA (contributed by
Barry Cooper of the Jet Propulsion Laboratory)

SITE: JOHNSON SPACE CENTER

Title	Type	Cognizant People	Hardware	System Software	Status	Function/User Community
Vibration Acoustic Lab	Custom	Walt West FTS 525-2517	Computer: PDP 11/34 Graphics: E&S/PS2	OS: RSX-11 Support software vendor supplied: SDR by Structural Dynamics Research High Level Command Language for kinematic display	Complete	Function: shuttle support orbital fatigue life analysis model, testing on critical areas of structure. Interactive analysis of test data, data base (modal shape tables), curve fitting, kinematic display.
Engineering Structural Analysis	Turnkey/custom	Joe Rogers FTS 525-3576	Computer: DPR-4 Graphics: Adage GS340 w/2 workstations Interactive w/host UNIVAC 1110 on communication link w/auxiliary display station - Tektronix to Univac 1110.	Language: FORTRAN Software support NASTRAN (McNiel-Schwindler)	Complete	Function: Structural analysis for space shuttle orbiter. Pre and Post-processing of NASTRAN I/O; original structure deformed structure dynamic display. Model builder; grid point and structure builder and editor.

SITE: JOHNSON SPACE CENTER

SITE: JOHNSON SPACE CENTER						
Title	Type	Cognizant	System			
		People	Hardware	Software	Status	Function/User Community
Simulation/ On-Board Graphics Display	Custom	Jim Smith Jim Van Arsdale FTS 525-4571	Computer: PDP 11/35 SEL 32/55 Graphics: Vector General 3400(2)	OS: RSX-11	Complete	Function: shuttle support
Color Scene Generator	Custom	Jim Smith Jim Van Arsdale FTS 525-4571	Computer: PDP 11/40 Graphics: E&S PS2	OS: RSX-11	Complete	Function: shuttle support

SITE: LANGLEY RESEARCH CENTER

Title	Type	Cognizant People	Hardware	System	Software	Status	Function/User Community
AD-2000	Custom	O. Storaasli B. Huff FTS 928-3401	Computer: Prime 750 VAX 11/780 Graphics: TEKTRONIX		AD-2000/IPAD	Complete	Automated design and drafting; video tape is available, contact BETH HUFF, training office 2611
Interactive Design and Analysis of Future Large Spacecraft Concepts	Custom	L. B. Garrett	Computer: Prime Graphics: Tektronix		LASS	Complete	LASS-Large Advanced Space Systems modeling for lattice (truss-like) structures and simplified multi-discipline design and analysis; executive data management graphics 20 application models: structural, thermal, control system modeling on-orbit static, dynamic thermal loading analysis, structural element design surface accuracy analysis and cost algorithms.
PICASSO	Custom	O. Storaasli FTS 928-3401	Computer: Prime Graphics: TEKTRONIX Ramtek		PICASSO	Complete	Program to Integrate Controls Aerodynamics systems and Structures Optimization
GFRAPO	Custom	O. Storaasli FTS 928-3401	Computer: Prime Graphics: TEKTRONIX Ramtek		GFRAPO	Complete	Graphite Fiber Research Advanced Project Office digitize and analyze composite fiber distributions from electron-microscope slides.
SPAR	Custom	O. Storaasli FTS 928-3401	Computer: Prime Graphics: TEKTRONIX Ramtek		SPAR	Complete	Supersonic Cruise Aircraft Research generate, display and modify complex structural finite-element models.

SITE: KENNEDY SPACE CENTER

Title	Type	Cognizant People	Hardware	System Software	Status	Function/User Community
Automated Drafting System(ADS)	Custom/ turnkey	Joe Burke FTS 823-4747	Electronic draft- ing table attached to minicomputer no I/O device	None	Complete currently looking for re- placement	User: drafting engineers for production drawings.

SITE: AMES RESEARCH CENTER

Title	Type	Cognizant People	Hardware	System	Software	Status	Function/User Community
CADAM Facility	Turnkey	Herb Finger FTS 448-6598	Computer/graphics Applicon 701 (Operational 7 yr) (3) stations Applicon 85 (Operational 6 mo) (1) station (1) station in FY'81		Vendor supplied	Complete	2-D electrical and electronic drafting (Planned to move to 3-D)
Computer Graphics Facility	Custom	Jim Hart FTS 448-6251	Computer: PDP 11/45 Graphics: E & S PS1 Network w/VAX CDC 7600 ILLIAC via ARPANET		DECnet ARPAnet FORTRAN	Complete	Display computational fluid dynamics w/contour and surface plots data analysis w/user-oriented manipulation; full rota- tion scaling molecular modeling (groups and sub- groups) user script driven movie (16mm) in b&w/ color on a D48 DIKOHED. Semi-real-time screening of data for time sequence analysis (air particles over wing)
Computer Graphics Facility	Custom	Jim Hart FTS 448-6251	Computer: PDP 11/40 Graphics: (2) E&S PS1			Complete	Aeronautics: preview complex geometry of air- craft (Coon's patch and bicubic patch analysis) menu driven 3-view Theoretical PHYSICS: modeling in physics re- search of galactic evolution 2000 stars, to 100,000 stars Stratospheric modeling: pollution over simulated land (movie).
Aircraft Aerodynamics	Turnkey	Jim Cozzolongi FTS 448-5855	System: CALMA DDM 1 station			In progress operational 8/1/81	Turboprop development project.

SITE: DRYDEN FLIGHT RESEARCH CENTER

Title	Type	Cognizant People	Hardware	System Software	Status	Function/User Community
Interactive Control Systems		Kevin Peterson FTS 984-8311 x357				Design stations for interactive control systems.

SITE: GODDARD SPACE FLIGHT CENTER

Title	Type	Cognizant People	Hardware	System Software	Status	Function/User Community
NASCAD	Custom	Lloyd Purves FTS 344-5837	Computer: VAX Graphics: TEKTRONIX (4016) RAMEK RATT	OS: VMS Language: FORTRAN	In Progress	Interactive system for input pre-processing and data definition, and output post-processing; plotting interfaces that permit the display of structure and substructures with or without deformations.
NASTRAN	Custom	Lloyd Purves FTS 344-5837	Computer: VAX	OS: VMS Language: FORTRAN	Complete	An all-FORTRAN version of the level 17.6 NASA structural analysis program NASTRAN. Two new features are: the program can be executed from a terminal in a manner that permits use of the VAX interactive debugger, and any link can be interactively restarted as often as desired.
APT/CAMI	Custom/ turnkey	Lloyd Purves FTS 344-5837	Computer: VAX Applicon	OS: VMS Application Software: CAMI	Complete	Direct link between detail design and CAM (manufacturing and assembly).

SITE: GODDARD SPACE FLIGHT CENTER

Title	Type	Cognizant		System		
		People	Hardware	Software	Status	Function/User Community
Structural & Mechanical Design	Turnkey/Custom	Lloyd Purves FTS 344-5837	Computer: Applicon 701 Graphics: (5) Applicon	Language: FORTRAN	Complete	Design studies of proposed Space Shuttle payload configurations: specialized software automatically calculates the mass properties of each mission configuration
Shuttle Science System Scheduling	Custom	Scott Lambrose FTS 448-8552	Computer: VAX 11/780 Graphics: TEKTRONIX RAMTEK AED512	Language: FORTRAN DI-3000 (High-level graphics package)	Complete	SPIRE project support: 2000 star system for science system scheduling for shuttle.

SITE: JET PROPULSION LABORATORY

Title	Type	Cognizant	System		Status	Function/User Community
		People	Hardware	Software		
Computer-vision facility	Turnkey	Craig Elliott/ Jim Ilix FTS 792-6627	Computer/graphics CAADS 3 Computervision w/4 workstations 1 CALCOMP PLOTTER 1 300 mbyte disk	CADD3 3 Software	Complete	System does 2 and 3-dimensional mechanical design and automatic routing PC boards, electronics schematics, forming of finite element models, instructions for numerically controlled machining and mass properties determination.
OPTICS DESIGN	Custom	Dr. James Breckinridge FTS 792-6785	Computer: PRIME 550 Graphics: (2) Tektronix 4025 disk 96 mbyte printer 300 LPM	Language: FORTRAN and commercially available system ACCOSV	Complete	CAD Tool to support infrared and advanced sensor system research design and development, use optical conceptional design for Infrared Astronomical Sattellite (IRAS). Upgrade: CAD radiometry and scattered light programs.
LSI/VLSI Custom Design	Custom	B. Cooper FTS 792-6159	Computer: VAX 11/780 Graphics: AED/512 Printer: TRILOG Plotter: HP	Language: FORTRAN SALOGS MP2D SPICE FETSIM LOGSIM ART	In progress to complete 10/82	A set of public domain LSI/VLSI design tools are being integrated with a relational DBMS with software development of fabrication house interface.

SITE: JET PROPULSION LABORATORY

Title	Type	Cognizant People	Hardware	System	Software	Status	Function/User Community
Computer Graphics Lab	Custom	R. Holzman B. Cooper FTS 792-2544	Computer: VAX 11/780 PDP 11/55 Graphics: E&S PS2 AED-512 E&S FB DEANZA FB		OS: VMS RSX 11	Complete	Research and development in computer graphics; i.e. animation. Completed tasks: Venus map- ping, ocean floor mapping, Voyager, VOIR, Pioneer 11, comet rendezvous movies, tank battle simulation, Seasat radar geometry In progress: preliminary design for spacecraft configuration and analysis preprocessing of antenna structures and finite element analysis.

SITE: LANGLEY RESEARCH CENTER

Cognizant						
Title	Type	People	Hardware	System Software	Status	Function/User Community
Supersonic Aero Branch	Custom	M. Adams D. Miller	Host: CDC 6600 Computer: PRIME 400 Graphics: TEKTRONIX	Quick	Complete	Quick- geometry package describes aircraft geometry in a form useable for numerical aerodynamic computations
Hypersonic Aero Branch	Custom	S. Stack W. Small	Host: CDC 6600 Graphics: TEKTRONIX Graphic tablet RAMTEK (color)	Action w/GEMPAK (geometry package)	Complete	A Computer-Aided Design System geared toward Conceptional Design in a Research Environment (ACTION) - a computer graphics display technique for the examination of aircraft design data - quickly sorts and inter- prets large amounts of data.
Vehicle Analysis Branch	Custom	A. Wilhite J. Rehder T. Rau	Host: CDC 6600 Computer: PRIME 400 Graphics: TEKTRONIX	AVID EDIN	Complete	Geometry and first order analysis AVID (Aerospace Vehicle Interactive Design) links a number of indepen- dent computer programs, each specializing in a particular technology, via communications data base - analyze vehicle geometry, aerodynamics performance mass/sizing, and cost.
IPAD Office	Custom	O. Storaasli Loendorf FTS 928-3401	Host: CDC 6600 Computer: PRIME 400 Graphics: TEKTRONIX Computer: VAX Graphics: TEKTRONIX	SPAR	Complete	SPAR-a finite element, structural analysis program that performs static and dynamic analysis of both linear and non-linear systems-stress and buckl- ing problems as well as vibrational analysis.

SITE: LEWIS RESEARCH CENTER

Title	Type	Cognizant People	Hardware	System Software	Status	Function/User Community
General Graphics	Turnkey/custom	Ralph Everett FTS 294-6163	Computer: IBM 370/3033 Graphics: ADAGE Cluster (1-4 terminal stations)	General graphics package; device inde- pendent in- house develop- ment, will use CAD/CAM, CADAM, or McAUTO	in- progress completion 1/82	Applications: computation- al fluids, circuit design; structural design- analysis w/NASTRAN I/O; facility engineering drafting.

SITE: MARSHALL SPACE FLIGHT CENTER

Title	Type	Cognizant People	Hardware	System Software	Status	Function/User Community
CADAT Large Scale microelec- tronics Computer Aided Design and Test System	Custom	Dr. John Gould Ted Edge FTS 872-3766	Computer: SEL 32/55 Graphics: Chromatics Hardcopy: XEROX-6500 (color)	Language: FORTRAN	In progress Completion date: 9/81	Users: LSI/VLSI designers Upgrade of place and route technology; CADAT system to include VLSI size data bases, HDL (High-level Design Language), and improved interactive graphics (AIDS)
Program Development Design	Turnkey	Carl Colley FTS 872-3067	System: Computer- vision CADD53 3 workstation Calcomp plotter card reader Disk (20 words) Communications to 1108. (1200 baud)	HASTRAN on 1108	Complete	Mechanical design; com- munications interface to 1108 to run HASTRAN for finite element analysis.
Structural Design	Custom	Arvin Hudgins FTS 872-2234	Computer: PDP 11/70 System: M&S/IGDS7 2 stations System: M&S/IGDS8 4 screen station w/3D capability plotter: Calcomp 960/1055s.	OS: RSX-11	Complete	Design/drafting for pro- pulsion structural and mechanical design. System has 2-D and 3-D capability with high- baud rate connection to minicomputer.
Electronics & Control	Custom	Bill Boglio FTS 872-4361	Computer: PDP 11/ 70 System: M&S/IGDS 20 Stations w/2 screen storage tube; TEKTRONIX hard- copy (8) plotters Calcomps 960/1055 Tablets versatic plotter/printer digitizing tables	OS: RSX-11	Complete w/GMA dual 19" raster tube up- grade	Preparation of electrical schematics, interconnect diagrams, panel layouts, and rack allocation drawings: multi-layer pcb designs; schedules; assemblies; parts in- ventory.

SITE: MICHoud ASSEMBLY FACILITY/NEW ORLEANS

Title	Type	Cognizant	System		Status	Function/User Community
		People	Hardware	Software		
Shuttle External Tank Assembly	Turnkey/ Custom	John Evers FTS 685-2572	Computer: VAX 11/780 Graphics: TBD		In progress 2 year/\$2 million program	Function: 40 station system to support shuttle external tank assembly. Proposed system may in- clude 8 VAX computers currently turnkey com- panies are each setting up a system for demo purposes.

Appendix D. The Need for Data Management Systems in NASA

Various NASA space missions have generated enormous amounts of raw data. A better understanding of Earth and space can be achieved by converting this vast resource into meaningful information using modern computer science and technology (CS&T). Computer science is therefore fundamental to the space sciences, a research discipline opened largely by NASA itself, and to many other endeavors requiring the management of large data streams. In deciding how to incorporate CS&T into its data handling operations most effectively, the Agency may view itself in least two different ways:

(1) NASA as a large organization and scientific establishment. In this capacity, the Agency needs to be run efficiently. For that purpose, it needs effective management information and decision support systems with automated offices. (See Appendices G and L.)

(2) NASA as the "Eyes and Ears of Spaceship Earth." In this capacity NASA collects, processes, saves, retrieves, and helps utilize information vital to future human survival and prosperity on this planet. NASA will require enormous computing power (and novel computer science techniques) to perform such a task. To date, computers have been exploited by NASA extensively but by no means fully.

Most commonly, NASA is regarded as requiring advanced computers for physical control of complex, semi-autonomous missions in space. Indeed, the first few NASA unmanned spacecraft used simple mechanical control devices. (One employed a wristwatch-type mechanism to shut off a relay system after one year of operation.) By contrast, each Viking spacecraft that traveled to Mars during 1975-1976 had a programmable computer aboard. Each was repeatedly reprogrammed to cope with unanticipated developments, thus enhancing the value of the mission many times.

But the technologies required to perform hardware-related objectives satisfactorily have been addressed elsewhere in this Report and are comparatively well-known. In this Appendix we briefly examine computer science and technology issues which are less apparent to the casual observer

-- in particular, problems related to management of large databases generated by NASA missions.

D.1 Efficient Large Data Management Systems

NASA as a data acquisition, management, and processing agency has no precise parallel in industry or any other government agency past or present. Many of its needs and problems are unique. In particular:

(1) NASA's scientific and applications missions generate huge volumes of data. In 1978 alone, more than 10^{13} bits of data were sent back to Earth from space. In the future this rate is expected to increase substantially (see Appendix E). Figure D-1 gives some idea of the expected single-mission data volumes. An exact assessment of the future volume cannot be made since the launch of these missions depends upon availability of budgetary funds and on final on-board sensor configuration, but in any case the expected future traffic is expected to be considerably more than that presently being handled by NASA.

(2) The rate of growth of data from scientific and applications missions has been nearly exponential. For example, 4000 LANDSAT digital image scenes were provided to the user community in 1980 as compared to only 10 in 1973. Any slackening of the pace of new data processing capability installation will cause backlogs to build up exponentially.

(3) Variability of data format from various missions is another NASA-unique problem. Since this potential information resource must be made understandable to the scientific user community, uniform data formats and easy access to databases is a pressing Agency need.

(4) The value of data lies in the amount of information that can be extracted from it. NASA has no choice but to depend upon other sources for such extraction. Unlike industry, NASA must interact with and educate a wide range of scientists and other users in its data processing operations.

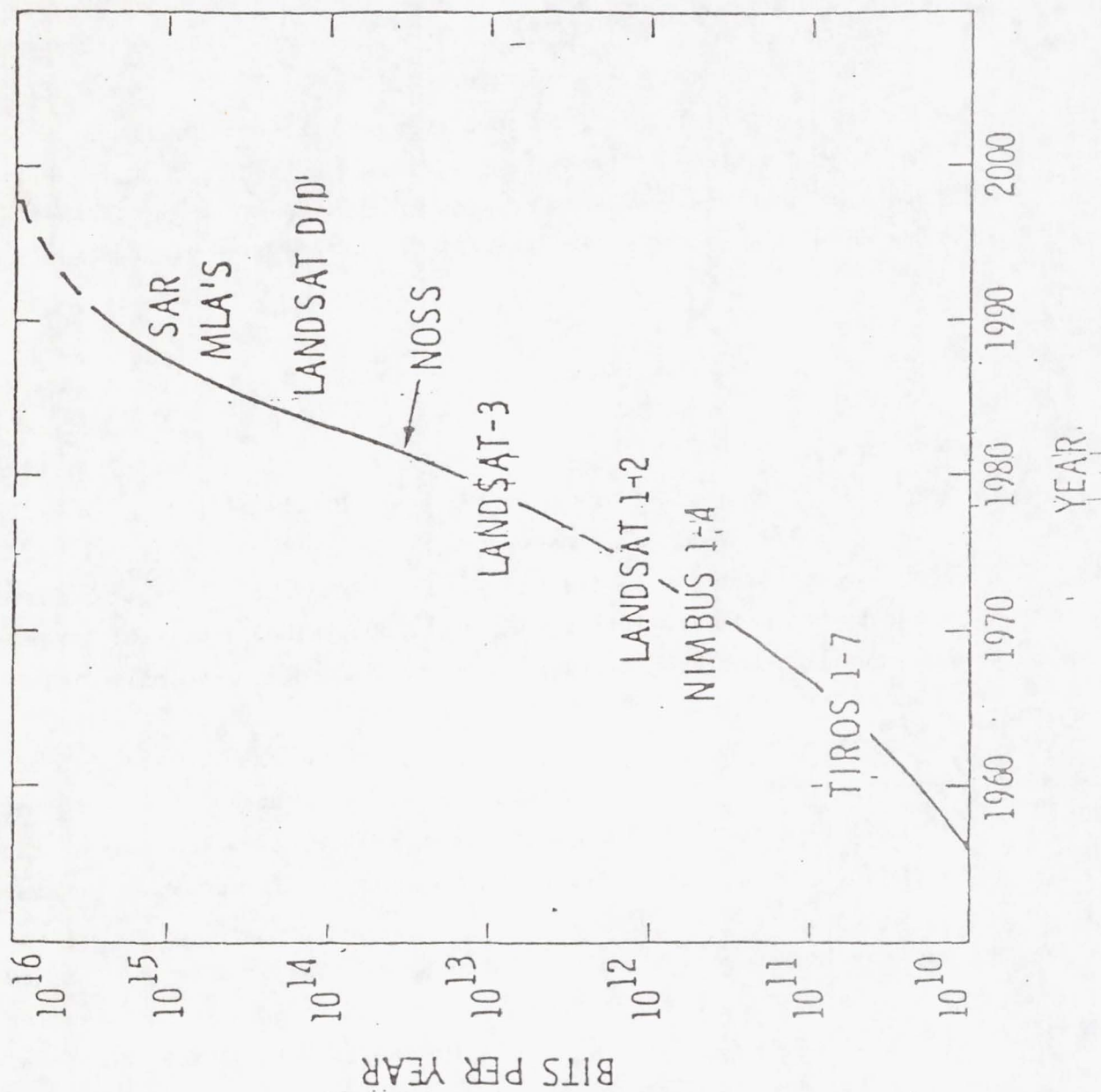


Figure D-1.

(5) In cases where rate of transmission of data does not match the rate of acquisition of data, or where acquired data is of insufficient quality to warrant transmission to ground stations, NASA needs on-board processing capabilities. The component worthiness, architecture, and data management problems associated with such "semi-intelligent" systems are unique by their very nature.

NASA's shortcomings in the area of data management have mostly been due to a lack of appreciation of the complexity of the problem rather than a lack of some specific piece of technology. Because of indirect public pressure, NASA's priority is to assure success of many missions in as short a period of time as possible. As a consequence, in the past, "housekeeping" activities were largely ignored. In retrospect, a number of observations can be made.

Most of NASA missions generate enormous volumes of data, but usually almost none of it is ever used -- a colossal waste of a valuable resource. For example, in 1978, 10^{13} bits of data were sent by various missions but only 10^{11} bits, that is, 1%, could be processed by the Principal Investigators (PIs). Further, NASA failed to adequately involve the scientific community in planned dissemination of information from projects before the launch of those projects. And the scientific community was seldom involved in data system planning for data from a specific mission.

Because of this lack of foresight, data systems and analysis activities were not properly funded. Underestimation of lengths of mission and cost of production of software have been the sources of this problem, which has been further compounded by transfer of allocated funding to other categories.

On-board processing of data has not yet been seriously attempted. In cases where the rate of transmission of data is slower than the rate of accumulation of data or data having no useful information content are acquired (e.g., presence of large cloud covers in case of crop identification models), on-board processing of data seems the only solution. With some modifications, on-board data handling is technically possible with the latest advances in LSI/VLSI technologies.

Data distribution problems have been experienced for many of the following reasons: Long delays between receipt of data at ground stations and delivery of processed data to users; high-cost data processing (e.g., \$200 per "scan" for LANDSAT images); failure of PIs to pass on raw data to other users (after they have achieved the satisfaction of first publication based on analysis of the data); improper cataloging and browsing facilities for users; lack of standardization of data formats, making it difficult for users to understand and correlate different data types; data archives contain insufficient information about the quality and limitations of stored data -- such as time, attitude, orbit and sensor correlation information; and the fact that some data are available only through PIs rather than a central agency.

Software problems have been caused by a number of factors, among them: (a) Software is generally developed by each PIs to suit his needs, in many cases resulting in duplication of effort and higher costs for all; (b) Software is not adequately documented; (c) Having been developed by PIs for a specific local computer, software is not easily "transportable" for use on other computer systems; (d) Software is often not written in a professional structured format; (e) No standard software language has been specified Agency-wide; (f) Software is commonly incomplete at time of mission launch, thus adding to the eventual processing backlog.

Another difficulty is that PIs must either depend upon local computing power (e.g., rented time or in some cases dedicated minicomputers) or travel to remote places to process data (which slows output per unit of time and increases dollar cost per unit of activity). Computational capabilities available have not kept pace with need.

There have been serious problems in the area of data storage and retrieval. It is common for insufficient storage space to be planned. To create space for new data, old data often must be purged -- thus material which might have become useful at a later date becomes instead a casualty of mismanagement. Tapes are the usual storage medium -- compared to optical and video disks, tapes require appreciable storage space and have a short life in terms of reliability. Further, catalogs of stored data are not widely and readily available. This causes serious retrieval problems.

The power of interactive-processing has not been fully utilized. In fact, no effort has been made by NASA to standardize languages and algorithm-definition. There has been insufficient attention to on-line processing of data. Intercommunicating, interactive terminals have not been widely used. Because of a lack of real-time processing, robots have not been fully exploited.

Finally, bureaucratic inertia has created a serious lag in the utilization of current available technology. Shortage of funds has also precluded the replacement of older "dumb" terminals by newer "intelligent" terminals.

D.2 Efficient Large-Scale Database Management

If NASA is to successfully cope with the problems of (near) real-time processing, efficient storage, retrieval and distribution of great volumes of data, the following steps are urgently indicated:

(1) Involve scientists and the rest of the user community sufficiently before the launch of a mission in the areas of planning, exchange, archiving, and retrieval of data and in software planning. NASA should consider establishing standing advisory committees consisting of concerned user and NASA personnel for the entire life of a mission.

(2) Each mission-devoted Committee may make plans for the free flow of data to users at minimal dollar and time cost, for the standardization of data formats so that data from other missions can be correlated, for supplying ancillary data (e.g., time of observation, attitude, orbit, sensor calibration) along with primary data, for oversight of software development in structured format and the preparation of adequate software documentation prior to launch of a mission.

(3) Upgrade computer facilities for each user. This could easily be achieved by making available a network of small mini/micro computers accessible to each user, connected by databus to a mainframe computer of sufficient power to handle problems not within the reach of normal user-devoted minis and micros. (Such a network may cost approximately 7-12%

more than a centralized system but would probably be worth it.)

(4) Upgrade data storage facilities. A changeover from tape to optical and video disk modes ultimately may provide relief from physical storage problems.

(5) Establishment of a standing Computer Technology Development Committee might help to ensure Agency-wide exploitation of the latest technology with reasonable speed.

(6) Greater attention to on-board data processing and data compression techniques.

(7) NASA should undertake a Public Education Program with the aim of alerting potential scientific and applications data users to the availability of and modes of data retrieval.

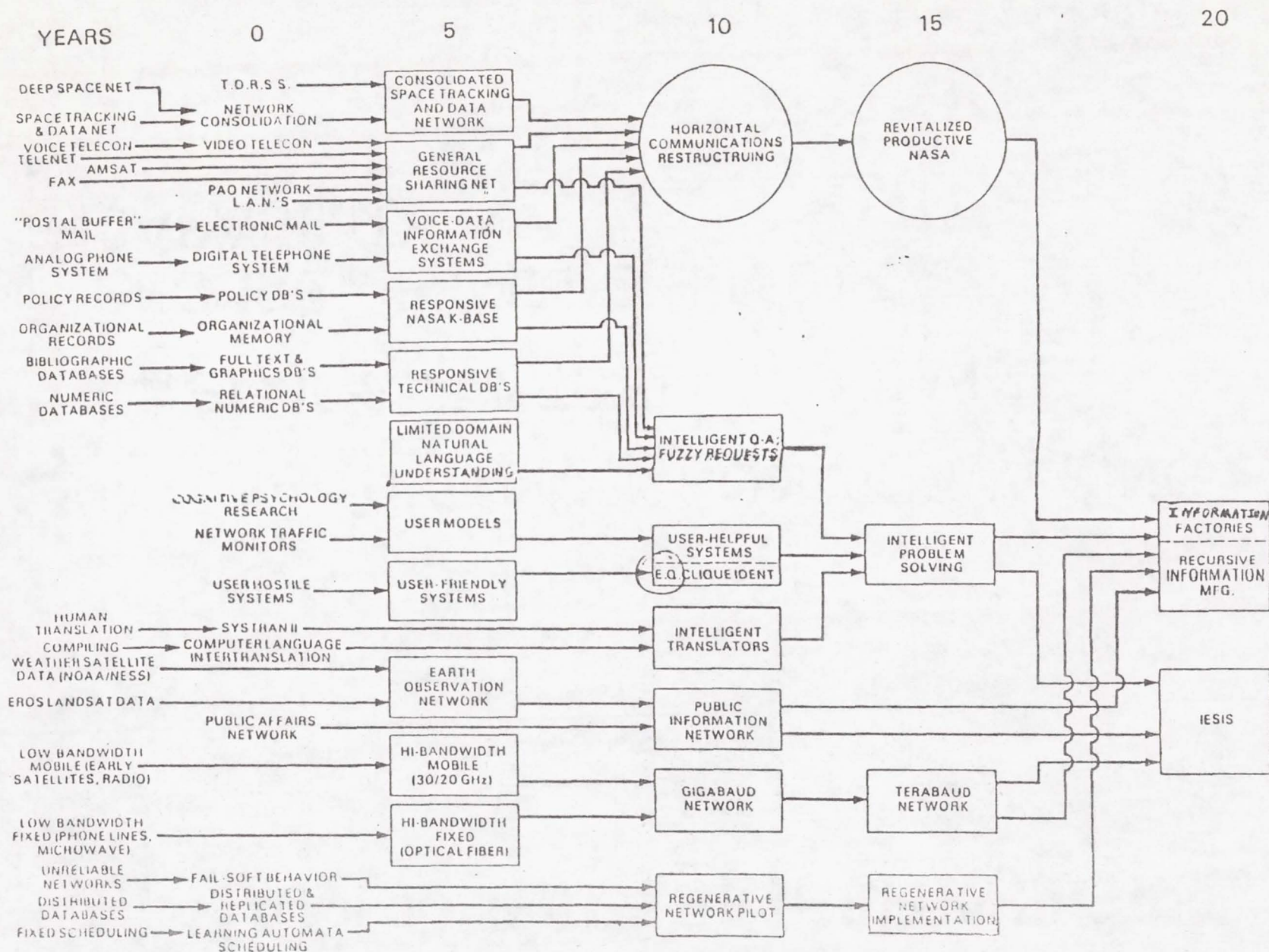
(8) A decade of neglect of "housekeeping" functions in data processing is finally catching up with the Agency. The time has come for NASA to commit serious funding to the areas of data management and software development, and to resist traditional temptations to transfer these funds to other areas in response to short-term budgetary pressures.

(9) NASA should consider viewing universities as "vendors" of small software development projects and data management programs. Even though it may be "easier" for NASA personnel to work with industry vendors, universities have the potential of providing better service at a lower cost, for small-size projects.

It is true that the Agency has begun to seriously address some of the issues raised above, as for instance the Applications Data Service (ADS) and the NASA End-to-End Data Systems (NEEDS) program. However, full implementation of programs such as NEEDS depends upon future Agency funding. In case of expected budget cuts and inadequate funding, NASA must resist the short-sighted "solution" of allocating funds to physically visible missions at the cost of "obscure" data management systems such as NEEDS. It makes little

sense to collect data if almost none of it can be processed in any reasonable time-frame.

NETWORKS & COMMUNICATIONS IN NASA: A PLAN FOR PHASED IMPLEMENTATION THRU GRACEFUL EVOLUTION



PHASED IMPLEMENTATION

RESPONSIVE DATABASES

INTERACTIVE QUESTION-ANSWERING

INTELLIGENT PROBLEM SOLVING

RECURSIVE IMPROVEMENT
CAPABILITY



computer communications in the private sector, Dr. Lewis M. Branscomb (1981), vice-president and chief scientist of IBM, has this to say:

One sign of this growth is the way computer terminals are pervading business and industry. Today in the U.S. work force, there is a computer terminal for every 48 employees. Among IBM customers, there is a terminal for every 25 employees. And within IBM itself, we have a terminal for every 5 employees. By 1986, moreover, it is estimated there will be a terminal for every 10 employees in the U.S. work force; one for every 6 of our customers, and one for every 2 IBM'ers.

The protocol used by all the companies, with one exception, was the protocol of the equipment suppliers. Database decentralization was the prevailing system used by the companies, especially where there is a large diversity of products and subsidiaries within the company or there is a large volume in overseas operations. Computer equipment in use ran the entire gamut of the computer industry, although IBM hardware clearly was the most popular. Where the company grew internally, the same equipment supplier was usually retained; on the other hand, conglomerates tended to have a wider variety of equipment suppliers. Electronic mail was surprisingly popular, with only one company reporting no use of it at all.

Most of the private sector personnel who were interviewed hesitated to predict the future, perhaps because they are so involved with the present. A few typical comments were:

- o Did not foresee voice activated terminals in the near future, if ever, because it's too slow and because of language barriers.
- o If voice activation was going to be used, it would occur primarily at the top management level.
- o Networks would reduce the need for large numbers of main line computers. The companies would just hook up to a network that had the computer capacity needed and not need one of their own.

However, the overwhelming impression, based on the assessment of the private sector, was that:

- (1) NASA and the private sector are both faced with approximately the same magnitude of growth, problems, diversity of operations, and new technologies.
- (2) The private sector has evolved their networks and communication systems in a coordinated and planned manner.
- (3) High-level management positions must be a part of the organization structure to control the systems.
- (4) The computer systems managers are planning for the future with the whole company in mind.
- (5) They are making use of the necessary technology systems outside the company when such technology and systems are not available within the company or are cost effective.

E.2 Near-Term Prospects for Networks and Communications

This section describes some networking activities which NASA might undertake more or less immediately, as natural and important first steps in the evolution of its information-handling capabilities.

E.2.1 Consolidated Space Tracking and Data Network

NASA operates two worldwide space tracking and data acquisition networks. These systems provide communications, command and control (C³) services for many scientific and applications-oriented space vehicles. One of these networks, the Space Tracking and Data Network (STDN), provides C³ services for spacecraft in Earth orbit, including translunar orbits. The other, called the Deep Space Network (DSN), manages planetary exploration spacecraft (Smylie, 1981).

STDN consists of twelve ground stations worldwide, much smaller than during the Apollo era. This network daily services about 40 spacecraft. The recently successful first mission of the Space Shuttle **Columbia** was supported by STDN. In its present topographic configuration STDN provides at least one communication contact per orbit with each satellite (Smylie, 1981). A replacement for the STDN is presently being developed, called the Tracking and Data Relay Satellite System (TDRSS). TDRSS will become operational in 1983,

and will consist of two operational spacecraft in geosynchronous orbit, in addition to an orbiting spare. Thus a spaceborne tracking and data system will assume the C^3 responsibilities of the ground-based STDN.

DSN consists of three ground station complexes at Goldstone, California, at Canberra, Australia, and at Madrid, Spain. Its deep space C^3 activities are managed by JPL in Pasadena, California. After TDRSS becomes operational, DSN and STDN will be merged under a network consolidation plan (Figure E-4). The resulting system will provide enhanced data return capabilities for planetary missions. New automation techniques also will be introduced into the network for the purpose of reducing operation and maintenance costs.

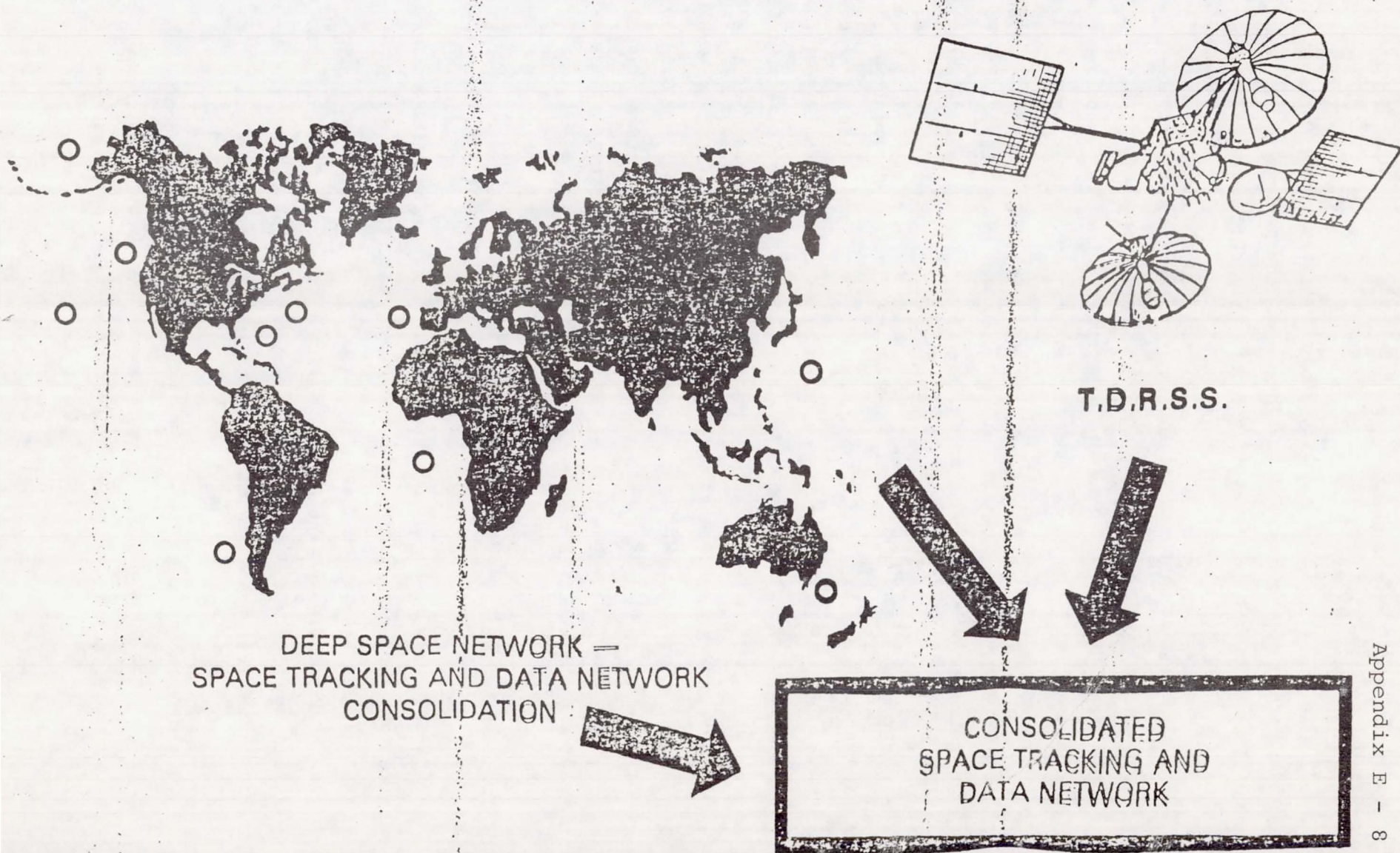
The trend of scheduled upgrades and continued consolidation is expected to continue for tracking and data networks. A new generation of tracking and data relay satellites might provide services to synchronous orbit vehicles, satellite-to-satellite communications, direct lines to processing facilities, and full translunar and near-Earth-to-Earth orbit coverage with no gaps. For the 1990's there are tentative plans for the development of an Orbiting Deep Space Relay Station (ODSRS). An outward-looking TDRSS, the ODSRS would eventually replace the DSN. Consolidation of these systems (including standardization of communication protocols) may be necessary to meet future mission and scientific requirements.

E.2.2 A General Resource Sharing Network

NASA uses a number of computer networks and information systems. While these are basically functioning satisfactorily, they are dedicated systems which are difficult to manage, are unable to communicate with each other and cannot be modified or extended significantly to meet fast-changing needs. Simple examples are FTS (Federal Telecommunications System) and the U.S. Postal Service, upon both of which NASA relies heavily.

A very short-term change which would increase NASA's contact with the computer science community and enhance resource sharing opportunities would be to place the Agency on the ARPANET using hosts at each Center and at

EVOLUTION OF TRACKING AND DATA RELAY



Headquarters. This calls for an accompanying decision to have a liberal ARPA access policy for Agency personnel, including the easy availability of terminal facilities. It also calls for administrative decisions concerning the NASA resources to be shared. RECON is certainly a likely candidate.

Such an action has several anticipated beneficial effects. First, it would make the human and machine resources of the external world more readily available to NASA personnel. For example, Agency personnel would gain easier access to such systems as MACSYMA and DIALOG. Secondly, it would give NASA people far greater access to whole communities of specialists functioning in high-technology, computer science-oriented activities such as artificial intelligence (AI) and numerical linear algebra research. Networking NASA to the external world would make the resources of NASA available to the external world and thereby increase the pool of specialists knowledgeable about NASA, its resources, and its problems. Finally, joining the ARPANET would permit the computer science community within NASA itself to remain in close contact with the external computer science community, thus permitting them to keep in touch with the most recent developments. Top-quality computer scientists regard this form of communication as fundamental to their activities.

Areas for experimentation with networks might include:

- o User resistance/adoption,
- o Congestion and control,
- o Standarization of protocols,
- o Distributed databases,
- o Distributed processing,
- o Remote mobile hosts,
- o Systems architectures,
- o Modeling and simulation, and
- o Reliability.

Network Monitoring

Contemporary research in electronic communications networks is primarily concerned with standardizing protocols, interfacing different systems, traffic

and congestion control, system reorganization in response to failure or congestion, tariff structures, and routing strategies. These are areas that relate to the structure and operation of networks themselves. However, the traffic flow contains information about the organization or system of organizations that provide the traffic. This kind of information has interesting potential, as illustrated by the following analogy.

Suppose the flow of traffic on the network of roadways of a city is monitored. Type of traffic, rate of flow, and congestion are tracked. From this information certain higher-level insights into the city, its economy, the kinds of activities that occur (and where and when they occur) can be derived. Furthermore, assuming sufficient flexibility to reorganize the city and to analyze the traffic flow in a very detailed way, this information might be used to "tune" the city structure (e.g., by rearranging the location of sources and sinks) so that its traffic flows more smoothly and uses fewer total miles.

It is this idea that may also be useful in streamlining traffic flow on a network. There are future areas for social network analysis of this type. For instance, by studying the content and flow of messages and information among individuals in a complex organization, the network monitor may be able to detect groups of people who have interconnecting activities and interests even before those individuals are aware of one another.

Document Modification Analysis

A complex system has a life cycle with many phases. In coarsest terms these phases include conception, design, implementation, and evaluation. In the case of an ongoing system this process becomes highly iterative, repeating itself in an irregular wave pattern perhaps with overlap and even interference of these waves of activities. Next consider a single cycle -- if one step is poorly implemented the next step becomes confused and inefficient. For example, if the conception is not well thought out and agreed to by the principals, the design phase will be chaotic. This will manifest itself in the appearance of confusion on the part of the designers. As the design process proceeds, it will be subject to excessive revisions because the

conception will still be in an iterative stage.

When a true understanding of systems is at last obtained, it will be possible to develop profiles of behavior of the design process that represent sound design. Then, by monitoring the development of design documents and the changes they undergo, it will become possible to detect weaknesses in either the process of developing those documents or in the conception of the project. This same reasoning applies to the relationship between design and implementation, and between implementation and evaluation. However, there are two very profound levels of research and understanding involved before this potential can be achieved. The first is a very thorough understanding of the development of a system, and the second is the generation of a set of standards for each step -- design, implementation, and evaluation -- if it is assumed that the previous step was carried out correctly.

Social Implications

While the above proposals may or may not ever be feasible or useful, perhaps an even more important question is whether they will ever be desirable. Clearly there are major questions of invasion of privacy. Whether the kinds of capabilities suggested above ever become reality or not, all communications networks carry with them implicit capabilities to detect and to accumulate information about individual users and groups of users which today cannot normally be accessed. For example, one of the corporations whose Communications Network Director was interviewed has installed a completely digital telephone system as part of his company's in-house network. The system automatically monitors the total usage of every telephone. This means that managers will now have access to certain information about the behavior of each employee which has not customarily been in management's hands.

Critical issues affecting the quality of personal and professional life will undoubtedly arise as communications networks become more sophisticated, commonplace, and far-reaching. Balancing the potential for effective management against the need for human dignity is a problem to be faced by society during the coming decades.

E.2.3 Voice-Data Information Exchange Systems

To achieve message transfer at a distance, manual postal services have been used since the Hellenistic Age. Only after the invention of the telegraph by Morse did an electric message handling service become generally available. Yet postal systems and telegraph/telephone systems have evolved along different paths, with major functional differences. Recent developments, however, indicate that this situation may change with the development of integrated voice and data information exchange systems.

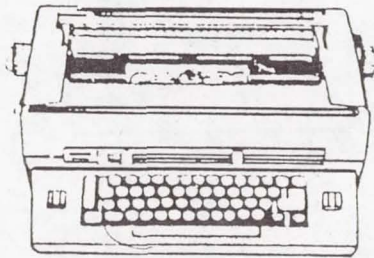
Traditionally, mail has been transmitted through a kind of "postal buffer." The postal system is called here a "buffer" because the effect of sending a message through the mails is equivalent to loading the message into a specialized buffer, there being held for an indeterminate period of time before it is read by its intended receiver. Electronic mail is a way of accelerating the buffer process to a point at which there is virtually no delay between message transmission and reception. With the new technology, messages typed, written, or drawn can be transmitted digitally at speeds approaching the speed of light.

NASA currently employs electronic mail service only on a limited basis. Many potential electronic mail users are less productive because they cannot transmit messages quickly enough. State-of-the-art electronic mail capabilities, such as those employed by some large corporations and universities, are not currently available to NASA employees. Of the electronic mail systems that are available to Agency employees, many, such as the Automated Information Management System are not user-friendly. Perhaps as a consequence, use of electronic mail has not yet reached the upper management levels.

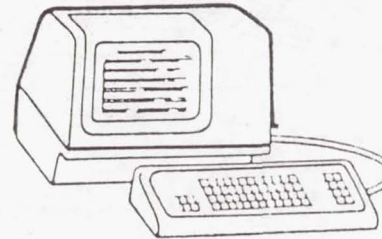
The age of analog telephone systems is giving way to one of all-digital phone systems. The telephone company plans to upgrade most of its lines 56000 bits per second digital. Some local phone systems are already digital. Besides the benefits of increased reliability, digital phone systems allow increased data packing, higher signal-to-noise ratios, digital transmission and digital recording of voice messages.

Figure E-5

EVOLUTION OF INFORMATION EXCHANGE SYSTEMS



"POSTAL BUFFER"
MAIL



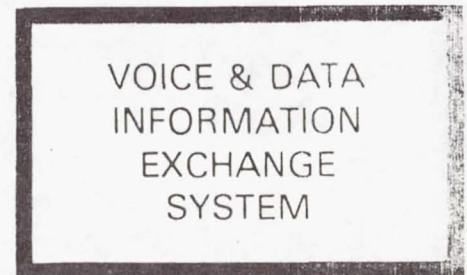
ELECTRONIC
MAIL



ANALOG PHONE
SYSTEM



DIGITAL PHONE
SYSTEM



NASA continues to rely on existing FTS voice-grade analog phone lines. The FTS system will not allow high data-rate transmission, increased voice quality, nor higher reliability. Digital storage of phone messages is not possible within the FTS system. With regard to digital telephone technology, NASA (and indeed, much of the U.S. government), lags well behind the commercial state-of-the-art.

The logical extension of contemporary electronic mail and digital telephone systems is an integrated voice-data information exchange system (see Figure E-5). Devices already capable of reading text, primarily designed for the blind, can be modified to transform mail messages into voice signals. It will then be possible to dial-up an electronic mail message service and hear messages. Branscomb (1981) has elaborated on other possible capabilities:

Developments such as speech filing and discrete word recognition will bring digitized voice messages into the system, as well. Like images, they can be referenced, edited, distributed, and archived.

More sophisticated networks will involve a variety of traffic types, including not only voice and alphanumeric data, but also digitized video and other imagery. The feasibility of integrating voice and data traffic in a broadcast using a random access scheme has already been demonstrated (Pan, 1978). In this scheme, silent moments in voice communication (typically 60% of total conversation time) are filled with data transmission. Channel resources are used only if there is a voice signal or data packet originating at one of the terminals.

E.2.4 Responsive Databases

One of the purposes in constructing a network is to share resources, and one of the most valuable resources, information itself, can be stored in databases at various nodes in the network. Networks thus offer opportunities for the sharing of information. At issue, however, are such questions as the relevancy of shared information, ease of access, and completeness of what is available online. Also the administrator, the manager, the scientist and the engineer each has special needs. Some of these needs are easily met with

RESPONSIVE SCIENTIFIC AND INFORMATION DATABASES

RESEARCH AREAS

DATABASE MANAGEMENT SYSTEMS
(RELATIONAL DATABASES)

DISTRIBUTED DATABASES

DISTRIBUTED PROCESSING

RESPONSIVE KNOWLEDGE BASE

POLICY DATABASE

NASA MANAGEMENT INSTRUCTIONS

ORGANIZATIONAL MEMORY

RESPONSIVE QUERY SYSTEM

currently available technologies, whereas others can be met only with projected extensions of these technologies (Figure E-6).

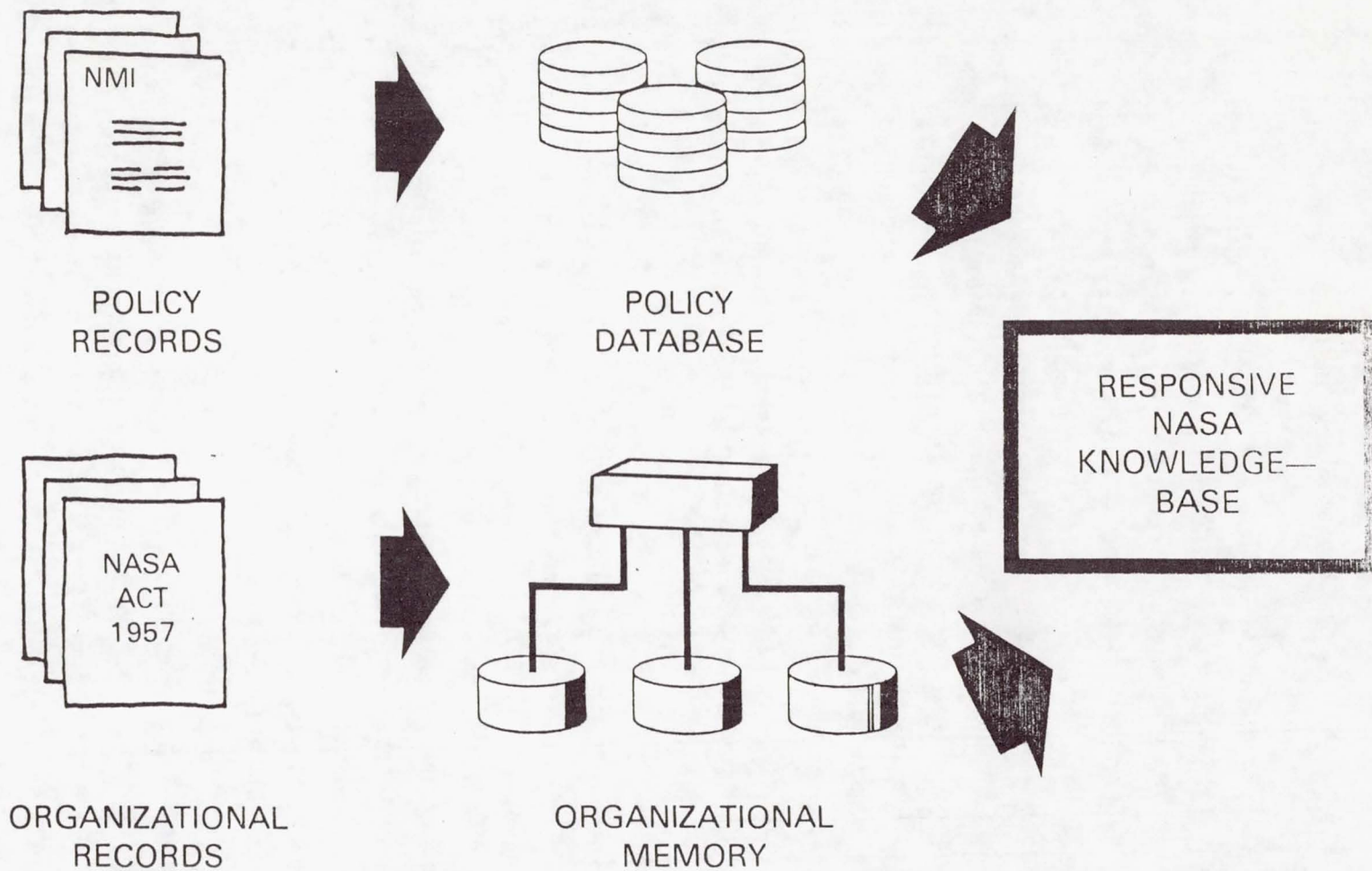
Full-text storage databases allow some companies to digitally store and retrieve entire policy documents on-line. In these companies, management instructions can be entered into a central database remotely, then transmitted to the relevant personnel via electronic mail. This centralized approach allows organizational personnel to review policy statements from a single source, through on-line computer terminals.

Within NASA, policy documents such as NASA Management Instructions are mass-printed and mass-distributed. Frequently they do not reach the people for whom they are intended. Storage facilities consist of nothing more advanced than filing cabinets. Retrieval of management instructions is wholly manual at present, and there is no automatic method for obsoleting or deleting old policy documentation. Agency administrators and managers concerned with the dissemination, retrieval, and storage of policy documentation within NASA are not taking full advantage of computer database capabilities which are already commercially available.

Contemporary database technology allows activities such as scheduling, message transfer, policy instructions, and case histories to be archived electronically as well. This database constitutes an organizational memory from which users can review the developmental history of projects. With a few keystrokes an employee accessing an organizational memory could know what has been tried, what works, and what does not. He can learn from the experience of others who have attempted similar tasks, and benefit from their mistakes and their successes (Figure E-7).

NASA presently has no computerized organizational memory. Since scheduling, message transfer, policy instructions, and case histories are neither transmitted nor stored digitally, the construction of an organizational memory is not immediately possible. As a result, locating organizational records is a time-consuming and expensive task. There is no reliable method for learning about past accomplishments and errors within NASA other than by direct communication with the individuals involved. The Agency has not used

EVOLUTION OF MGMT. INFORMATION



database technology to create an organizational memory.

NASA pioneered the development of bibliographic databases with its Remote Console (RECON) system. RECON accelerated the research process considerably, inasmuch as it completed literature searches in a fraction of the time required by manual searching. Today, however, increased disk storage capabilities, higher communications bandwidths, and faster processors allow the construction of full text and graphics databases. Since much technical writing and publishing is accomplished using digital equipment (e.g., word processors and electronic typesetters), it is a comparatively trivial task to load full text into databases. Digitizers allow drawings and illustrations to be stored digitally as well. Optical character readers would permit printed matter to be entered without time-consuming manual keypunching. Entire technical articles, not just bibliographic citations with abstracts, could be stored electronically and searched automatically in full text and graphics databases.

NASA-RECON and other databases within the Agency continue to rely upon older standards for data storage. Time delays persist even within current NASA databases. RECON, for example, provides lists of references with abstracts. There is still a time delay during which the technical articles themselves are retrieved. Patent licensing records have only recently become available through computer databases to NASA patent counsels, yet patent drawings and full records must be retrieved from filing cabinets. With existing technology, these and other databases could be upgraded to handle full text and graphics.

Numeric data bases have proliferated in recent years. New developments in relational databases have allow numeric databases to become more than just lists of numbers. Specific relations inherent in raw data can be stored in the databases also. Data are retrievable not only sequentially but also according to some specific relation. NASA Langley Research Center has experimented with relational databases such as RIMS (Relational Information Management System) for retrieving data on the Space Shuttle's Thermal Protection Tiles.

A natural extension of existing database technology is the intelligent, or responsive, knowledge base (Figure E-8). Construction of full text and graphics databases is proceeding already; NASA is likely to follow suit in time. Relational numeric databases such as RIMS have been developed by NASA. Policy data bases and organizational memories are already appearing in the private sector. NASA may choose to develop these as well.

A responsive NASA knowledge base would contain a variety of management information (Figure E-9). It has absorbed the policy database, containing such items as the NASA Management Instructions, as well as the organizational memory, containing such items as case histories. Augmenting the database is a responsive query system. Characteristic of this query system is a uniform set of commands capable of accessing all data, whether distributed or local. These commands will resemble natural language as much as possible. The database is responsive in the sense that it can interact with the user to narrow or broaden query requests and to obtain more specific search definitions. If data are gathered from more than one database source, this need not be apparent to the user. A responsive NASA knowledge base will allow managers to quickly and easily access the wide variety of data needed to manage efficiently.

A responsive technical database is an extension of full text and graphics technical databases and relational numeric databases. Like the responsive NASA knowledge bases, it can interact with users to better define search problems. The query language is uniform for all data stored. A responsive technical database, properly designed, will allow NASA engineers and scientists to gather research data much more quickly than was ever possible before. The productivity of NASA researchers can increase dramatically.

Responsive or intelligent knowledge bases can, like many communications and information processing technologies, reduce needless paperwork. Data will be gathered as required. Hardcopy need be only an occasional by-product of an otherwise totally electronic, on-line information system. Users will become more familiar with the computerized tools over time, and query languages will become increasingly transparent to users. The responsive knowledge base will make practical laborious research which was not practical before, and it will

accelerate current research processes.

E.2.5 Limited-Domain Natural Language Understanding

Some general problems of artificial intelligence are made explicit in the domain of natural language understanding. Knowledge representation, symbolic representation, and "meaning" are typical problems faced by natural language theorists in artificial intelligence. A somewhat stronger position is that making a computer understand natural language insures artificial intelligence, just because the same general problems apply. No existing computer has achieved language understanding over a domain comparable to the size and scope of natural English. Some progress in machine understanding of natural language has nevertheless been made.

Computer recognition of natural language is a highly complex task. Most researchers have asserted that it is best to begin with a limited domain in order to bound the problem. Success in the natural language area has indeed been greatest when highly local subsets of natural language are isolated as the domains of discourse. Typical examples of success in this area are the Lunar Sciences Natural Language Information System (LSNLIS) (Woods et al., 1972), SHRDLU (Winograd, 1973), and MYCIN (Davis, 1977). LSNLIS knows how to talk about moon rocks. SHRDLU converses about a world of colored building blocks. MYCIN is an expert on certain types of blood disease. Within limited domains such as these, computers can understand natural language questions and requests.

Algorithms such as the ATN (Augmented Transition Network) (Winston, 1979) make possible the generation of computer commands from natural language input. Once a method is discovered for associating natural language requests and questions with lower-level formal commands, the computer can automatically generate commands that it can understand. Certain types of unambiguous inputs are very easy to handle. The computer is less likely to understand the input if it is ambiguous. For instance, "The tall man the ships" and "Time flies like an arrow" are classic examples of ambiguous sentences beyond the scope of a computer program's capability at present. Simple and limited-domain commands and requests may be understood by contemporary computer systems.

Natural language understanding has many implications for communications and network systems. NASA is currently developing very little in this area. The bulk of the research in limited-domain natural language understanding has been taking place in university and corporate research laboratories.

E.2.6 User Models

User hostility of networks and their supporting services is a major impediment to the growth of network systems today. To use a contemporary network one needs to know a number of singular pieces of information just to access the system. While a knowledge of passwords and code words will always be required to safeguard costly resources from unauthorized or illegal users, much of the prior knowledge demanded of today's user is unnecessarily obscure. In an ideal environment a user should be required to know very few things in approaching the system through an interactive terminal -- perhaps just a key name to get into a query system and his own codename if the objective is to use some limited access systems.

Once into the query system, the user should be able to seek out the services required by consulting a heirarchy of menus consisting of no more than 10 items each. Because this may require many levels of menu, the query system should keep track of the descent through the menus which the user calls up. This track would tell users which option they chose on each menu and what the name of the menu was. User should then be able to roll back to any menu on the list, alter the choice at that level, then start a new descent. This is all a rather straightforward kind of user friendliness.

Developments which are not so straightforward, but which should be sought, are in the area of user friendliness based on user models. The idea here is that the query system would have a built-in expert system on the nature of users and their knowledge profiles with respect to the system being accessed. As the user attempts to access the system or obtain information from it, the system would be building a model of the user. This model would inform the system of the apparent level of sophistication of the user and enable a response at a level and in the manner most likely to be informative. This would make the system responsive to the naive user in a helpful manner but

sophisticated enough to please the knowledgeable user.

The motive for the above idea is quite simple. When a consultant in a highly technical field is asked a question, he forms a model of the sophistication of the questioner and creates an answer in terms and at a level appropriate to communicate based on the model formed. As the discussion continues, the consultant updates the model on the basis of the exchange. If the questioner turns out to be highly knowledgeable, the consultant greatly accelerates the discussion, uses technical terms more freely and makes references to well-known results and people in the field. If the questioner turns out to have little or no knowledge, however, an entirely different set of responses are crafted. And, of course, if the questioner is too ignorant, the consultant may deflect the questioner to some lower level source.

The consultant has many different types of information available to assist him in updating his model of the questioner, some of which cannot easily be made available to a query system. Such factors as body language, eye contact, rate of response, choice of terminology, and reasonableness of questions and statements are difficult to specify formally. Initially, query systems might be able to update user models on the basis of the nature and structure of responses, evidence of false starts, and so on. With more sophisticated systems and considerable research and development, rate of response and pace or consistency could perhaps be taken into account. Body language and eye contact will be far more difficult to implement. But it is altogether conceivable that a highly sophisticated system would look at a user as it is being used, and might even have sensors throughout the chair to detect signs of discomfort, fatigue, or impatience. Perhaps someday, query systems may tell human users to "take a break" in response to the following information: User has been at the terminal for 82 minutes, posture has begun to slip, user's eyelids are drooping, response time has fallen one-half a standard deviation during the last nine minutes after showing steady improvement for 32 minutes followed by essentially constant pattern, and the last three queries have been inappropriate to the intellectual level of the user. If the user insists on staying at the terminal, the system begins to degrade the model to a less-sophisticated user level.

E.2.7 High-Bandwidth Fixed Optical Communications

There is an ever-increasing demand for efficient communication systems, brought on by increasing costs to supply more new equipment and greater demand within NASA for all information services especially in telecommunications and computing. Reliance on the equipment and technology of the past decade creates many economic problems, the most important of which are increased costs for metal cabling, energy, and available space on existing transmission lines and on available carrier frequencies. Various new forms of telecommunications are appearing which require large bandwidths, placing a considerable burden on existing lines. The use of ultra-short wave links is rapidly approaching its limits in applications such as conference television, video telephones, and viewdata systems, in part because of a shortage of useable bandwidth and mutual interference. An innovative solution which emerged during the 1970's was to investigate optical frequency carriers (10^{13} - 10^{15} Hz) which advances in laser technology have made possible (Boraiko, 1979).

A shift to optical communications will, in time, provide as much as three orders of magnitude improvement in information bandwidth per channel. Other advantages of optical transmission links, as compared to conventional systems, include the following:

- (1) Fiber optic filaments 100 microns in diameter can replace a copper wire cable 10 centimeters in diameter, with equivalent capacity.
- (2) Optical transmissions are immune to ambient electrical noise, ringing, echoes, or other electromagnetic interference, and do not generate any of their own electrical noise.
- (3) Optical fibers are safe to use in explosive environments, and eliminate the hazards of short-circuits common in metal wires and cables because the fibers can be given total electrical isolation. Optical fibers are less failure-prone at elevated temperatures than coaxial cable. Furthermore, properly designed optical transmission lines and couplers are relatively immune to adverse moisture conditions and hence may be used more

readily underwater.

(4) Fiber optic cables today cost already about the same as premium-grade coaxial cable per unit bandwidth. As production volumes increase, fiber costs will drop to below 50% of their present values. Further, installation costs of optical cables are lower than for metal cables since shipping and handling cost are about 25% that of metal cables and labor is about 50% less (Elion and Elion, 1978).

Field trials have been performed to evaluate lightwave technology and to test cabling techniques, splicing techniques, and fiber optic equipment under field conditions. A 44.7 Mb/sec digital transmission system was put into operation by the Western Electric and Bell Laboratories in Atlanta in 1976 (Bell System Technical Journal, 1978). Similar field trials have been performed in England at 8 and 140 Mb/sec by the British Post Office (Berry et al., 1978; Murray, 1978), and in Berlin, Germany, at 34 Mb/sec by the Deutsche Bundespost (Liertz et al., 1978, Adler et al., 1978). Optical cables in the 8-280 Mb/sec range are already sufficiently developed and will soon be fully competitive economically with coaxial cables (Russer, 1980). At the Heinrich Hertz Institute in West Germany, 8-560 Mbit/s links have been investigated and developed, and an experimental 1.12 Gbit/s transmission link has already been used to carry 16 simultaneous 70 Mbit/s television channels (Baack et al., 1979). According to a representative from a major U.S. aerospace corporation, 20-fiber optical cables are commercially available today at the 44.7 Mb/sec/fiber level, for a total cable capacity of 894 Mb/sec. The technological capability for 144-fiber cables already exists, and 90 Mb/sec fibers are expected by 1982. This would place the state-of-the-art of optical fiber cabling at about 13 Gb/sec per cable by next year. A further doubling in per-fiber capacity is anticipated by the late 1980's in the commercial sector.

E.3 Mid-Term Prospects for Networks and Communications

Present and near-term networking activities can be expected to undergo phased implementation and evolve gracefully in such a manner as to obtain at least some of the advanced capabilities described below.

E.3.1 Intelligent Question-Answering

The simplest kinds of queries are those in which the boundaries of the problem to be solved are extremely well-defined and familiar. For example, a Los Angeles, California, user might give the following command to his local information outlet: "Book me on the next available flight to Washington D.C."

The machine would simply retrieve the proper airline protocol from its protocol directory, call up the airport computers and secure the necessary information. After asking whether you wish to fly into Dulles or National air terminals in the D.C. area, it might check whether an earlier flight might be obtained out of a Los Angeles area airport located farther away than the one nearest to your known address. It might also inquire whether cost, meals, movies, or other considerations are factors of interest, and if so what weight should be given to each. Finally, it would book your flight with appropriate airline computer. Such a system, except for the natural language interface, is probably within state-of-the-art (Winston, 1979; Buchanan, 1981). For instance, software which can interpret and explain the workings of an expert system database -- such as the MYCIN knowledge base tutor GUIDON (Clancey, 1979) -- are presently being studied.

Also under study are computer systems which can provide easy access for nontechnical personnel to large, distributed databases of information (Sacerdoti et al., 1978). The purpose is to develop mechanisms for automating many of the detailed tasks normally performed by a decision-maker's technical staff. This includes accepting a question in natural (but not necessarily grammatical) English, in the decision-maker's own terms; planning a sequence of queries to various files to gather the requested information; developing the plan into a computer program or programs in the language of the database management system on which the needed data resides; transmitting the retrieval programs, and monitoring their execution; and, finally, composing the retrieved information into a suitable output format. An early version of this system, called LADDER (Language Access to Distributed Data with Error Recovery), carried out all of these functions in at least rudimentary form. System and user performance have recently been evaluated (and improvements recommended) by the Navy for possible use of LADDER-like systems in a military command control environment (Hershman et al., 1979). A improved version of

LADDER, called SODA, has extended capability to access a heterogeneous database consisting of both Datacomputer and DBMS-20 database management systems (Moore, 1979). An experimental French-language version has been implemented as well.

For intelligent question-answering, a system must be able to understand "fuzzy requests" (Ke, 1979; Zadeh, 1979; Faria and Simoni, 1979). The following is an example of a poorly-defined request: "I need to find an article I just heard about, written by someone named Smith, or Smythe, or some similar-sounding name, published sometime in the last five years in an IEEE publication, on the use of fiber optics for nationwide networking." The system would recognize that "find an article" was the operative command, and would know the various ways of finding literature much like any modern, well-trained librarian. The first step (after verifying/confirming user input) would be to assume the user's information is correct as stated. Thus the system would search locally available computer catalogs of published articles in the title, subject, abstract, publisher, and date fields to retrieve and intersect all references keyed to "Smith" or "Smythe", IEEE, 1976-1981 publication, and the subject headings "Fiber optics," "nationwide" and "networking".

If the user's information is insufficient or not quite correct, one of two things will result: (1) Many items are returned from the intersection, among which may or may not be the item sought, or (2) no items survive the intersection.

In this first case, the search parameters must be further restricted because too much was captured. For the system to do this will require significant amounts of AI, natural language processing capacity, and an "intelligent" relational database query capability. The system must perform at least two tasks. First, it must attempt to create a "user request model," a relational model of what it is the user wants. (One experimental system, called EMYCIN (Van Melle, 1980), provides some assistance in structuring a person's knowledge about a problem by analyzing the attributes and relationships of the main objects in the domain of discourse.) This model is matched to the relational indexing scheme of the database itself, and then

another series of intersections is performed to pare down the list. After each iteration the system might scan the relational characteristics of the remaining items, asking the user something like: "Are you most interested in X, Y, or Z?" -- where X, Y, and Z are relational categories. The system will have subsystems which optimize this game of Twenty Questions, so that the fewest questions need to be asked of the user to identify the desired datum. Also if items can be separated on more than one basis, those choices having the greatest potential payoff should be asked first, with the user having the option of stopping the machine at any time to see what has been found. This is a very powerful approach: A series of only 20 binary responses can select the correct item from a field of 220 1,000,000 items. The main requirement is that every database item be absolutely distinct, preferably in numerous ways, so that a binary choice strategy will always work.

After a complete binary choice search, perhaps the desired item has not been found. In this instance, the situation is logically equivalent to having found no pertinent references, and so the procedures outlined in the second case become relevant as the next step of the search.

In this second case, the search parameters must be broadened because the item wanted was not found. This may be accomplished in one of two ways, perhaps using systems for deductive question-answering currently under development (Reiter, 1977), as follows.

First, each of the search indices may be explored and expanded in ways which are relationally reasonable from the "user request model" point of view. This may involve trying different configurations of the search terms used, trying synonymous search terms that have not yet been used, trying alternative relationally equivalent terms for the search. Or, the terms may be generalized to higher levels of abstraction (or greater distance from the specific) and a new search performed. All of this is accomplished interactively with the user, but at a far higher level of abstraction than in any system currently available.

The second avenue for expanding the search is to expand the search base itself. If the desired item is not in the local system database, then perhaps

it is available on NASA's RECON system, Lockheed's DIALOG system, The Source, or some other knowledge base. However, each new resource should be phased in gradually, according to some hierarchical ranking of probable success which also takes proper account of such matters as relative cost, speed of access, scope of materials available, and other relevant utility factors.

Once the search methods of case (2) have been applied, the user may again be confronted either with a large list of items possibly containing the desired item, or again, no items. In the former, we return to the methods of case (1) with the enlarged list. In the latter, the system determines that it has exhausted all normal library channels and must try other options. For instance, it may use a direct-dial capability to call a resource person or local computer at IEEE to find if they know about the article. (The limited natural language capability of the system would include information about how to connect with and converse with both humans and machines, including leaving messages on people's terminals and conversing with "computer secretaries.")

The system might search a master listing of college universities to retrieve departments or names of individual scientists whose area of expertise or interest is in the area of "networks" or "fiber optics." If negative, such a search could be expanded to master lists of industry or government scientists. If this approach fails, then the system might try pulling author/institution data from relationally similar articles located in the earlier search, and contact them for information. The challenges of coordinated problem-solving using many diverse and independent knowledge sources (Erman et al., 1976) and of content-directed translation and information mapping among machine-readable databases (Cahn and Herr, 1976) are already beginning to be addressed.

In the mid-term period of development, information systems should be capable of seeking out and gaining access to almost any source of information required by the user. However, it is still up to the user to apply the information acquired to some useful end.

Since the system has access to broad sources of information, it presumably also has access to the software "literature" as well. For example, perhaps

the user requests the mathematical integration of a particularly complex algebraic function. Rather than attempt to locate a publication which lists the integral already worked out (which may not be possible in many cases), the system might connect to the on-line MIT symbolic manipulation system MACSYMA (Moses, 1971), enter the formula for integration, and then report the result to the user. Thus the system would spare the user having to know how to connect to the MIT system, the login procedures and command structures. This seems appropriate, since the user cares only about the result, not in how it was obtained (except to verify that it came from a reliable source, which the system duly reports to the user.)

The information system may also give users direct access to various expert systems. For instance, a researcher in geology might wish to access PROSPECTOR, the expert system developed at SRI International to locate and map ore bodies by interacting with a nonspecialist geologist (Duda et al., 1978). In this instance, the researcher will know enough geology to understand PROSPECTOR's queries, and the system will act simply as an automatic hookup system. Prototype dialog programs that communicate with an expert to provide some of the same help that a "knowledge engineer" (who helps create an expert system) provides, such as TEIRESIAS (Davis, 1976), are now being looked at (Wood, 1981).

However, nongeologist users may also require information pertaining to location of ore bodies. In this latter case, the system will formulate a user request model and itself perform the interrogation of the specialized expert system. Since PROSPECTOR's queries are intelligible only to a person familiar with the jargon of geology, the system will spare the user this prerequisite and will extract the needed information and pass only the results along to the user. Naturally, the user may request explanations of how a conclusion was arrived at (Buchanan, 1981), but for the lay-user this level of confirmation will rarely be required. In most cases it will be sufficient to report to the user that his request was processed by an established, reliable system (Shortliffe, 1976) -- again, the user wants to know the answer to a specific question, not full details of how to solve problems of a certain general type.

In addition, it is expected that the system will provide access to

computational facilities as required by the user to interactively generate new information. Resource-sharing networks have already made their debut at the local level, and a few low-bandwidth nationwide systems are already in operation. In theory, the system should be able to connect a user with any computer, subject to various necessary access restrictions on specialized equipment such as high-speed mainframes. Ultimately, it should be possible to provide users with a "universal computational" capability in Turing's sense, thus permitting a more sophisticated user to program in any computer language he chooses. (Less knowledgeable users may request self-paced learning courses on specific programming languages, or may make use of software development tools including automatic programming systems (Barstow, 1979).) Note that the local system needs only the capacity to access such capabilities -- it need not be internally available.

In the next two decades, a wide variety of computer systems, interactive databases, and expert systems may be expected to come on line (Rosenberg, 1979; Shrobe, 1980; Wilkins, 1981). Already there are software packages which can play Master-level chess and countless other entertaining games (Samuel, 1959; Greenblatt, 1967; Waterman, 1970; Wilkins, 1980), diagnose medical diseases (Shortliffe, 1976; Buchanan, 1981) and comprehend 80% of all internal medicine (Pople, 1977), analyze the structures of complex organic molecules from mass-spectrogram data (Lindsay et al., 1980), prove theorems (Nilson, 1977; Stickel, 1978), analyze and debug electronic circuits (Sussman, 1975; deKleer, 1979), and even write computer programs according to formal specifications (Barstow, 1979). Many of these are not yet on-line for network access, but there are no serious technical barriers in the way.

Eventually a wide variety of problem-solving packages will become widely available in specific domains of human activity. This hypothetical system would store the callup procedures, protocols, capabilities, and other pertinent information in a resources database which it could consult in an attempt to match resources with user requests. The system resources database would also contain a listing of directory databases -- in effect, other "sources of sources" -- which it could consult in the event a user request for problem-solving could not be satisfied directly by the resources immediately at hand. After exhausting this approach without success, the system might

report that the requested information is not presently available. A regular search as described in the previous section might then be implemented, ending perhaps with the system recommending various helpful literature items or the names of people who might be able to assist the user in solving his problem. Such systems would require learning capabilities well beyond state-of-the-art in expert systems (Schank, 1976; Mitchell, 1977; Buchanan and Feigenbaum, 1978; Buchanan et al., 1978; Dunning, 1979; Hayes-Roth et al., 1980).

E.3.2 Intelligent Translators

NASA has supported the development of SYSTRAN and SYSTRAN II, a commercial machine translation system (Halliday and Briss, 1977). SYSTRAN II is a software package with 500,000 lines of code, together with a computerized dictionary which contains terminology, technical expressions, grammatical rules and semantic principles. The text to be translated is fed into the computer, where it is analyzed for syntax and semantics, then printed out in the target language. The computer's draft is refined by human translators, whose editing procedure is also computerized. The system then produces a magnetic tape ready for photocomposition. It is claimed that SYSTRAN II will "generally increase the output of a human translator by five to eight times, thus affording significant cost savings by allowing a large increase in document production without hiring additional people" (Haggerty, 1981).

SYSTRAN II applications include translation of service manuals, proposals, planning studies, catalogs, parts lists, textbooks, technical reports and educational materials. The system is operational for six language pairs -- English/Russian and vice versa, English/French and vice versa, English/Spanish, and English/Italian. Six other pairs -- English/German, English/Portuguese, English/Arabic, German/English, German/French, German/Spanish have been successfully demonstrated. English/Japanese and vice versa are currently being developed.

The ultimate goal is to be able to sit down at a compute terminal anywhere in NASA, perform data entry in any format, program in whatever style pleases the user, and then have that work accessible throughout the entire organization's resource sharing network. Programs should be able to run on

any part of the system where the data they require are available. Other people can run the same program if they too have the data. Any kind of data may be retrieved from any source accessible to the network. The net may also be used for electronic mail, audiovisual teleconferencing, and for storing, generating and manipulating all manner of telemetered, graphic, tabular, textual, and audiovisual materials (Figure E-10). Finally, as an ultimate goal for the future, the NASA network system should be accessible in natural language and capable of performing translations among human languages and between human languages and machine codes fluently and in real time (Figure E-11).

A fundamental problem is proliferation of computer hardware and the nonuniformity and noncompatibility of equipment, programming languages, and natural languages. How can NASA personnel work most effectively using equipment which is not designed to communicate with other equipment already in service? Machine-to-machine communication is analogous to the problem of trying to fit a square peg into a round hole. The solution is to devise a mechanism which can transform the programs, data structures and languages of one machine into those of another, while preserving implicit algorithms. The proposed device is called an Intelligent Translator.

There are a number of very basic difficulties which need to be examined prior to implementation of such a device. For instance, some systems use 36-bit words while others use 16-bit words. Thus one problem is how to preserve identical decimal accuracy between two systems during translation. Also, some systems allow a positive zero and a negative zero, while others have only a true zero, creating the problem of how to preserve the meaning of these characters. Perhaps we should question the necessity of using standard telecommunications protocol between two operating systems. Ideally two computers would be tied together with a translator in the middle, with no worry about how the data is going out of one and into the other.

When the above problems are solved and the aforementioned systems are implemented, the result is an intelligent NASA computer network. Full network control requires distributed processing. Also, intelligent translations between machines eventually may make it possible, for instance, to write a

FORTTRAN program on one machine and to translate it into a Pascal program for use on another machine. Compilers are the simplest kind of intertranslation between machine languages, but systems have been developed which translate structured programming text into FORTRAN (JPL, 1975; Melton, 1977; Osterweil and Myers, 1980), one version of ALGOL into another version (Martin, 1971; Kuo-Petravic et al., 1972), XPL into ALGOL (DaSilvamartins, 1978), IMP into PL/I (Nonweiler, 1974), and GOAL to HAL/S (Stanten and Flanders, 1974). Dickey (1979) has suggested a simple scheme for constructing translators between higher level programming languages. His approach, though not sufficiently sophisticated to construct arbitrary translators, probably is adequate for constructing translators between similar computer languages. Other "metalinguistic" translation systems have also been discussed in the literature (Michels, 1978).

The net would be accessible in a variety of languages, as well as natural language ultimately, for manipulating all the data within the intelligent NASA network system. There is some work being done in the area of "data translators" (Fry and Merten, 1977). Users may wish to access any NASA data, any NASA program, any NASA terminal, and select any input/output mode with which they feel comfortable. The net may be used for teleconferencing and accessing NASA film sources, ultimately with online real-time translation of natural languages (Van Berggeijk, 1978).

Such a system will have benefit both to NASA Project Managers and to people in the field. Within each NASA Center, all the computers will be able to communicate with one another. This capability may later be expanded to include other compute systems, various databases, and other nets external to the Agency network system. All Centers can retain their individual and diverse computer facilities without change and without forced standardization either of hardware or of software. The net will provide compatibility with all existing hardware -- full Center autonomy can be preserved, nurtured, even enhanced. The distribution of labor among the Centers is a matter for NASA Headquarters to decide, but undoubtedly each Center will have different functions and unique computing needs. This individuality can be maintained.

The intelligent NASA network can be implemented in stages, so that with

each new part that goes into the system some significant benefit accrues. The first component to be implemented should probably be the substations at each Center (in a distributed network scenario). After these prove themselves effective at the Centers for internal matters, the substations can be connected to one another via trunk lines. Finally, the capacities of the entire system can continue to be upgraded incrementally, with new capabilities initiated systemwide as they become available. There is no need to wait for the entire system to come on line to obtain significant results. It may also be necessary to construct at least one communications and control center, or a number of smaller substations netted together in a distributed fashion.

There should be a meta-algorithm for examining computer operating systems, then formally modeling them in an algorithmic fashion to break out the appropriate transition rules for translation into a "universal" network language. This might be some form of higher-level programming language which itself allows the description of an algorithm-oriented language. FORTRAN, for example, is designed to describe algebraic formulas; COBOL has business-oriented statements; the HAL/S programming language used for the Space Shuttle processes spacecraft control instructions.

Like the design of the HAL/S language, the network language might be high-level and specific to understanding computer operating systems. All languages could be translated into the "Intelligent NASANET" language, then translated out again. A single such "universal" meta-algorithmic language might not be necessary. Consider that there may be 100 different computer dialects requiring intertranslation on the net. If you create an intertranslation algorithm for every pairwise combination, 20,000 separate algorithms are needed. On the other hand, by using a NASA network language the number of distinct algorithms can be reduced to 200 -- 100 algorithms into the universal language from any language, and another 100 to reverse the process -- two orders of magnitude fewer algorithms.

It is true that inefficiencies may be introduced by intertranslation -- a strong argument for trying to use the same or similar language as the source language. But if this cannot be done or is not desirable, the Intelligent Translators on the net can transform existing software into a language that is

more compatible. The system should be able to provide some analysis of the validity of the translation and an assessment both of the probable errors and efficiencies that potentially may be introduced during intertranslation. Of course, the process could work the other way as well -- a programmer could write a program in BASIC which, upon translation into FORTRAN might turn out to be more efficient than before.

Development of Intelligent Translators would simplify ground control or make executive supervision of spacecraft easier because communication between ground and space systems would be facilitated. NASA would operate more effectively as a government agency because it would possess vastly greater software resources and more efficient access to that information than ever before. Future advances in computer science and technology could easily be infused into the Agency in a reasonably non-threatening way, since each person in NASA could retain and use his/her own personal computer system to speak into the network and to receive information from it. This would eliminate any need for users to spend great effort on worrying about compatibility or communications interfaces. Ultimately, natural language might be used in some applications.

A few possible directions for future research have been tentatively identified as follows:

- o It has been estimated that 95% of all programming code is written in various versions of FORTRAN. Practical implications of code-to-code translations in a FORTRAN-dominated programming environment should be examined further.
- o In addition to programming code translators, Job Command Language (JCL) translators are also required and should be reviewed.
- o Translation from ALGOL, Ada, LISP or Pascal Programs to COBOL, FORTRAN, or BASIC is inherently challenging. Problems to be addressed include the difficulties of translating recursive procedures, complex data structures, storage generation (memory allocation in COBOL, FORTRAN and BASIC is static), and parallel processing.

- o The effects on real-time programs should be considered -- will they still be "real-time" after translation?
- o Review formal and informal network theory as it might apply to the deployment and interconnection of Intelligent Translators.
- o Methods of integration using intelligent, machine-code-intertranslation systems of existing NASA information systems (e.g., RECON, LANDSAT data processing, and IPAD) should be considered.

E.3.3 Public Information Network

To stimulate public awareness of the space program, NASA could provide aeronautics and space data in a format ready for delivery to a general audience. The Agency is charged with the task of disseminating information about itself to the public (National Aeronautics and Space Act, 1957). One possible way to accomplish this goal would be to distribute public education material through modern commercial and public communications media.

As new network and communication technologies become available to the public at large, NASA could employ these media for the distribution of data to the public. In the United Kingdom, the commercially available data distribution system PRESTEL provides home computer enthusiasts with daily access to thousands of pages of data. Systems of the PRESTEL genre are becoming available in the United States. NASA could submit aeronautics and space data for distribution through these public data systems.

At present, toll-free telephone numbers at John F. Kennedy Space Center can be called to receive recorded information about upcoming launches. One way to upgrade such a system would be to offer toll-free telephone numbers for public information databases. Home computer owners could dial-up and search the public hotline databases.

In some parts of the United States, cable television companies have constructed systems which offer up to 60 channels of programming. No cable television company has yet been able to fill all sixty programming channels.

Public information about NASA could be distributed through now-empty cable TV channels. As a possible extension of this capability, the Agency should examine the prospect of distributing public information through direct-broadcast satellites. New television technologies can be exploited for the transmission of text, images, and films.

Another possible medium for disseminating public data is the home computer itself. NASA could make available aeronautics and space data on machine-readable disks and tapes. Many home computers have the capability to read eight-inch floppy disks. Others read inexpensive cassette tapes or smaller disk formats. The use of microcomputers continues to grow dramatically, a fact which invites pioneering efforts to distribute public data through magnetic storage media for home systems.

Thus far, public data distributed by NASA has consisted of publications, lithographs, slides, films and the like. All could be distributed through electronic media. NASA's only Agency-wide newsletter, NASA News, could be made available on-line and through other broadcasting media such as cable TV. NASA News has an extremely small distribution, yet its contents are of vital national interest. The information in this publication, along with other public education material, could be mass-distributed through modern communications media.

In the same way that weather satellite data has proven to be a useful supplement to daily televised weather reports, so it may be that LANDSAT and other remote-sensing data could be interesting to home viewers. Although remote-sensing data is not currently available on a real-time basis, future systems are expected to decrease turnaround times. It is reasonable to think of home image processing, image enhancement, or simple image viewing. Home viewers could find remote sensing data useful in a number of ways. Coastal residents could keep in touch with local sea and sealife conditions. During times of danger, they could keep abreast of hazards such as oil spills offshore. Urban residents might find useful such remote sensing data as pollution monitors, traffic reports, and urban development maps. Farmers have always found remote sensing images to be valuable crop management tools. Timely, up-to-the-minute information about insect infestations and crop and

RESPONSIVE TECHNICAL DATABASE

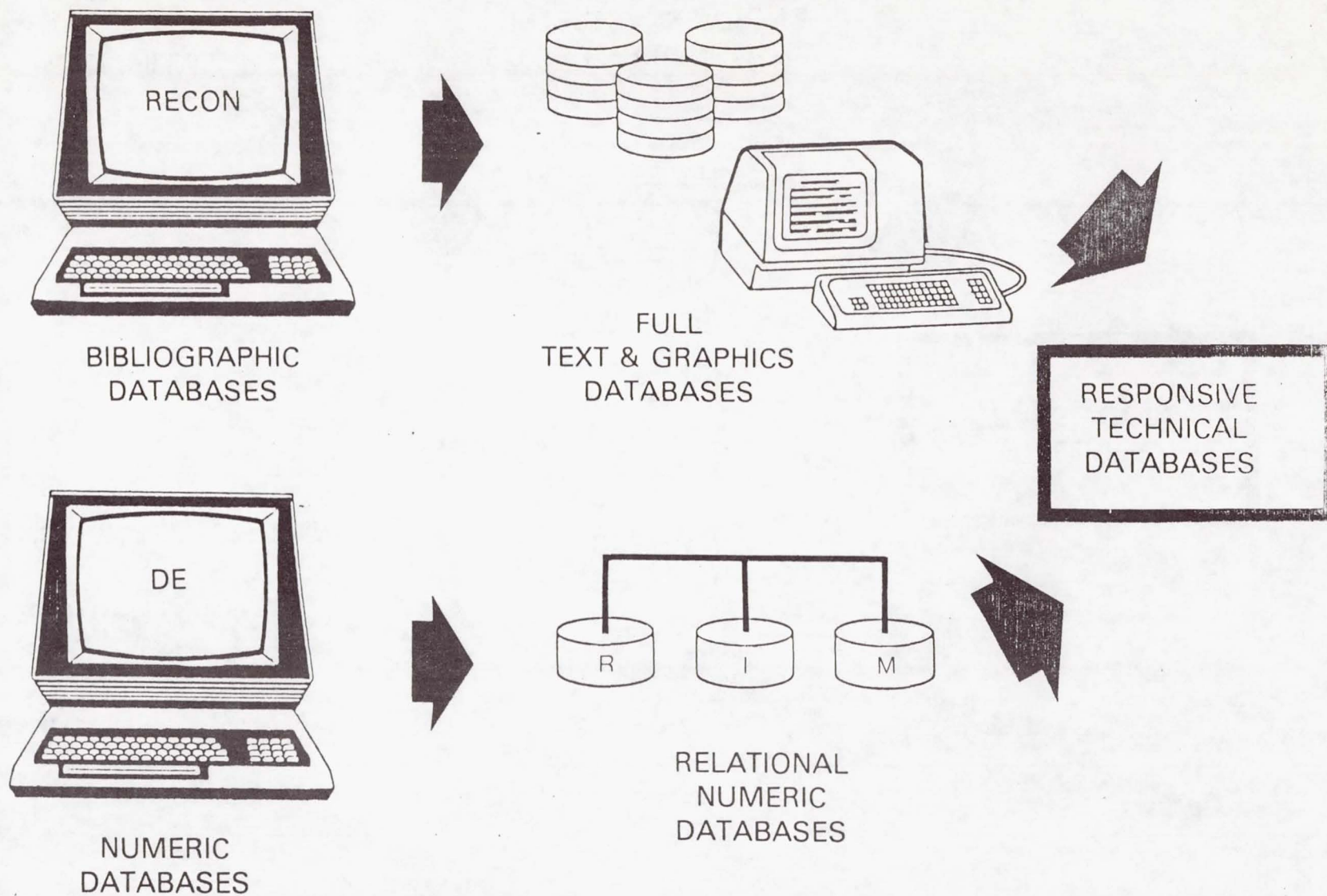
FULL TEXT AND GRAPHICS
TECHNICAL DATABASE

RELATIONAL NUMERIC DATABASES

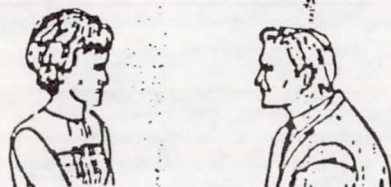
RESPONSIVE QUERY SYSTEM

INTEGRATED SCIENTIFIC AND
ENGINEERING DATABASES

EVOLUTION OF TECHNICAL DATABASES

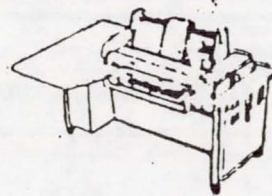


EVOLUTION OF TRANSLATION



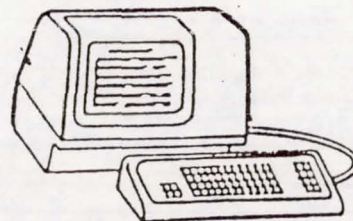
HUMAN TRANSLATION

訝 yaayn	amated
訥 neeyi.	niliwi
訟 Soongyn	Sullifig



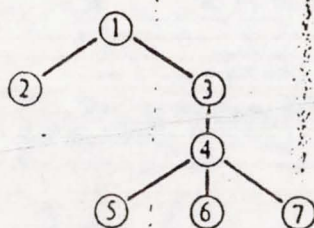
SYSTRAN II

A1AC 73 000C	1431
A1AF 96 05	1441 DELY
A1B1 44	1451
A1B2 D6 03	1461
A1B4 54	1471



COMPUTER LANGUAGE
INTERTRANSLATION

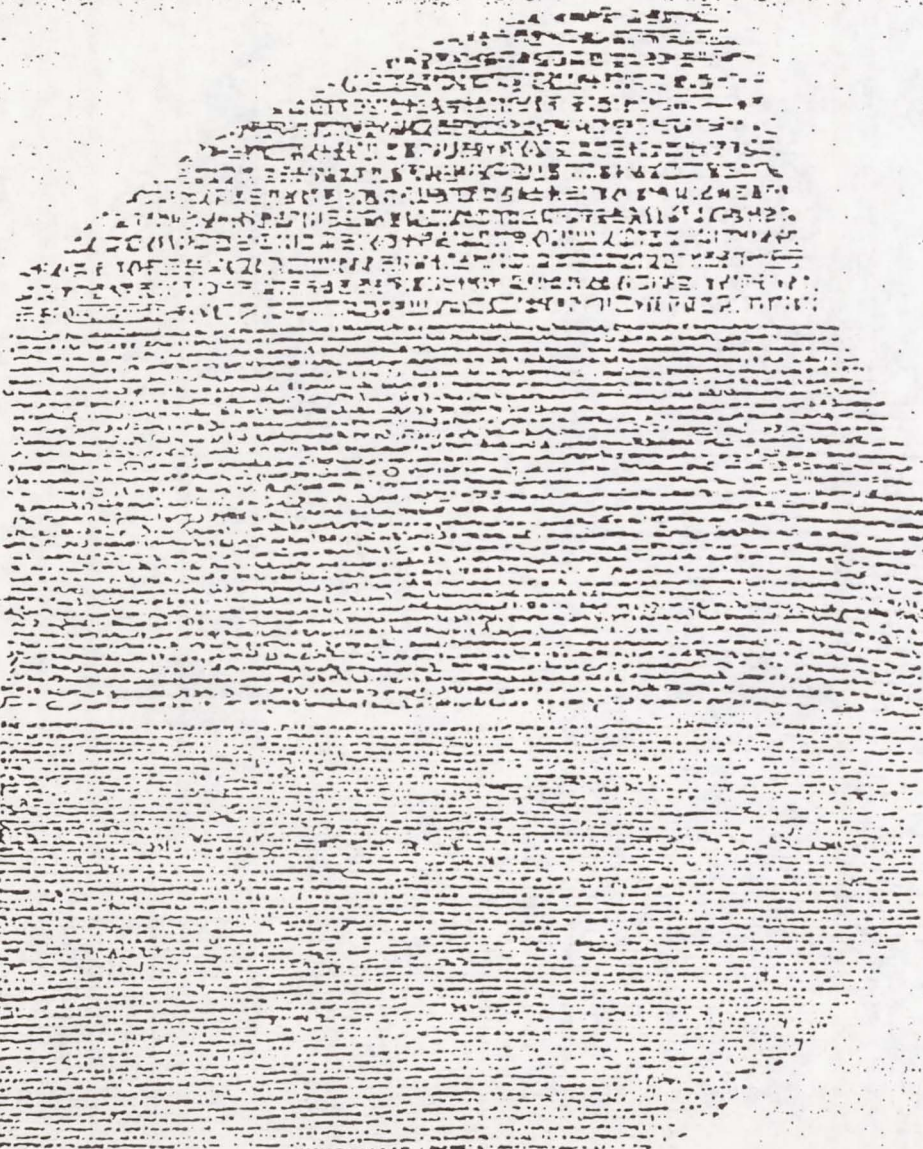
"INTELLIGENT"
TRANSLATORS



COMPILING

AUTOMATED NATURAL LANGUAGE TRANSLATION NODES: A MODERN ROSETTA STONE

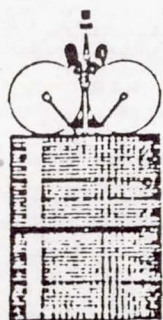
- MULTILINGUAL TELECONFERENCING
- UNIVERSAL NETWORK ACCESS
- TECHNICAL AND SCIENTIFIC LITERATURE XLATION
- INTERNATIONAL CO-OP SPACE MISSION COORDINATION



EVOLUTION OF HI-BANDWIDTH

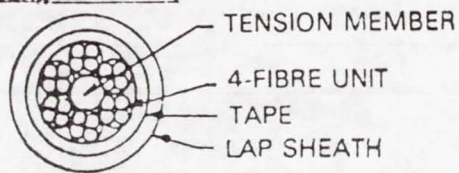


LO-BANDWIDTH
FIXED

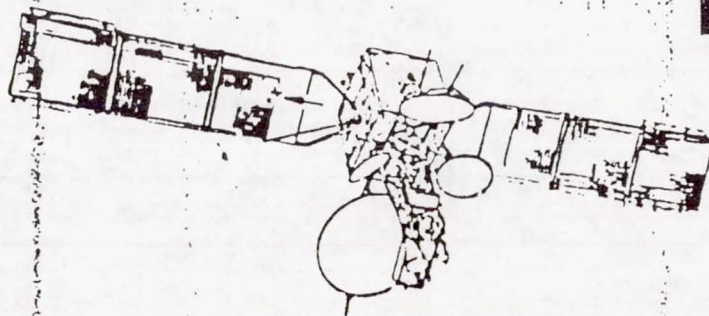


LO-BANDWIDTH
MOBILE

(d) 4-FIBRE UNIT CABLE



HI-BANDWIDTH
FIXED



HI-BANDWIDTH
MOBILE



soil conditions could make the difference between a healthy harvest and a blighted crop. During emergency conditions such as flooding and earthquakes, remote sensing data can help authorities manage an often uncertain situation. During the eruption of Mt. St. Helens, for example, high-altitude aerial photos provided disaster crews with maps of dangerous "hot spots." Even private citizens could benefit from such information -- as for example, when they must be made aware of rising flood waters. Finally, LANDSAT data has served a variety of special-interest users. While it is impossible to assess the impact of widespread distribution of, say, geological information, there is undoubtedly some portion of the population who would find these images valuable.

E.3.4 Gigabaud Networks

Besides the introduction of fiber optic links into existing telephone networks, this technology can be used for wide-band digital networks with full integration of data, voice, video telephone, and broadcasting applications (Figure E-12). A model of such an integrated network, with optical fiber links from 140-560 Mb/sec, was built at the Heinrich Hertz Institute in Berlin (Haller et al., 1978; Fussgaenger et al., 1979; Baack et al., 1979). In this test system, wideband signals are received from user stations via digital optical transmission lines at different transmission rates. User services include telephone, video-phone, television-radio, and communications. The optical lines, and the digital switching and multiplexing facilities with which they are combined, form a new type of digital wideband network with decentralized switching. The German group has also experimented successfully with a 1.2 Gb/sec transmission link. It is their contention that laser-glass fiber channels will displace all other digital transmission media.

In Japan, the government has cooperated with local communications firms to develop what may be the next generation of cable television (Nakahara et al., 1978; Hara, 1977). Called HI-OVIS, the system uses a single fiber for each on-line television set. Each subscriber is connected to the system by two fibers (one for upstream and one for downstream video transmission), and can access local news, shopping information, train schedules, and other information specifically related to the locality. Programs are based on

information selected and edited by the local inhabitants and system users. By using home-installed cameras and microphones, coupled with home keyboards and mobile broadcast centers, individuals can participate in a wide variety of programs. Home polling services can be used to obtain preferred programming information and to express collective interest in selected political, administrative, and community affairs. HI-OVIS provides a complete two-way communications service, including police, fire, and medical protection, multichannel CATV, and interactive and educational audio-visual programming. The basic home unit costs just a few hundred dollars to install.

An analogous system for NASA might involve installing an in-house optical fiber cable network capability at many NASA Centers. The typical NASA office today has a dial telephone as the sole means of communication with the outside world. The installation of an integrated local-area fiber optic service network could do much to move the Agency in the direction of increased organizational self-awareness. This might also forge a bond of organizational loyalty that would serve to increase productivity and an enhanced perception of personal relevance for NASA employees. Such a system would also make more feasible proposals like electronic mail and video conferencing because the necessary bandwidth for these projects would be available.

Another possibility is free-space optical laser links between satellites. The Air Force Space Laser Communications (LASERCOM) program started with system concept and component design in the early 1970's (Roland and Whited, 1978). By 1973, the communications system that has evolved demonstrated data rates up to 1 gigabit/second with a bit error rate of 0.0001% for 40,000-kilometer simulated links (Gardanier and Ryan, 1973). A six-phase demonstration of the LASERCOM system outside of the White Sands laboratory environment is presently under way. This demonstration will include ground-to-ground links up to about 20 kilometers and aircraft-to-ground up to about 50 kilometers, dynamic far-field acquisition, tracking, and two-way communications utilizing an orbital test payload to be launched in 1982 aboard the Air Force P80-1 spacecraft (Ross et al., 1980). Evolution to satellite-to-satellite links is envisioned in the future for the Air Force system, and research and development is proceeding in this direction throughout the communications field with capacities expected to be in the

15-20 Gbit/s range (Kennedy, 1974; English, 1980).

E.4 Far-Term Prospects for Networks and Communications

In the more distant future, perhaps around the turn of the century, NASA's requirements in the areas of networks and communications will be very different from those prevalent today. The following is a brief look at some of the future needs and technologies which NASA may require to perform its mission in the year 2000 A.D. and beyond.

E.4.1 Terabaud Networks

Even if one considers only those missions and applications programs already planned, the currently projected future bandwidth is barely adequate to meet Agency needs, and it is certain that the useful output of these missions will be significantly constrained unless additional bandwidth is somehow made available. However, when one considers the possibility of additional activities which are necessary if NASA is to achieve the goals outlined in Chapter 2, the projected communications capability is woefully inadequate.

Such "additional activities" are various functions which NASA is not presently performing, but which may be essential if NASA is to thrive during the coming "Information Age." These may include, but are not limited to, electronic mail, full on-line text editing and review, on-line real-time bibliographic search and full text retrieval, integrated voice-video-data conferencing, maintenance of a policy and administrative "organizational memory" accessible in a user-friendly content-relational format, and very-large-scale on-line data analysis. These are all high-bandwidth activities, and will require an Agency-wide communications capability far beyond that which is now available or anticipated.

The question of how much bandwidth may be required is very difficult to answer, but the attempt must be made to get some handle on the magnitudes of the quantities involved. Fortunately, there are three different approaches with converging results.

ACCESSIBILITY REQUIREMENTS

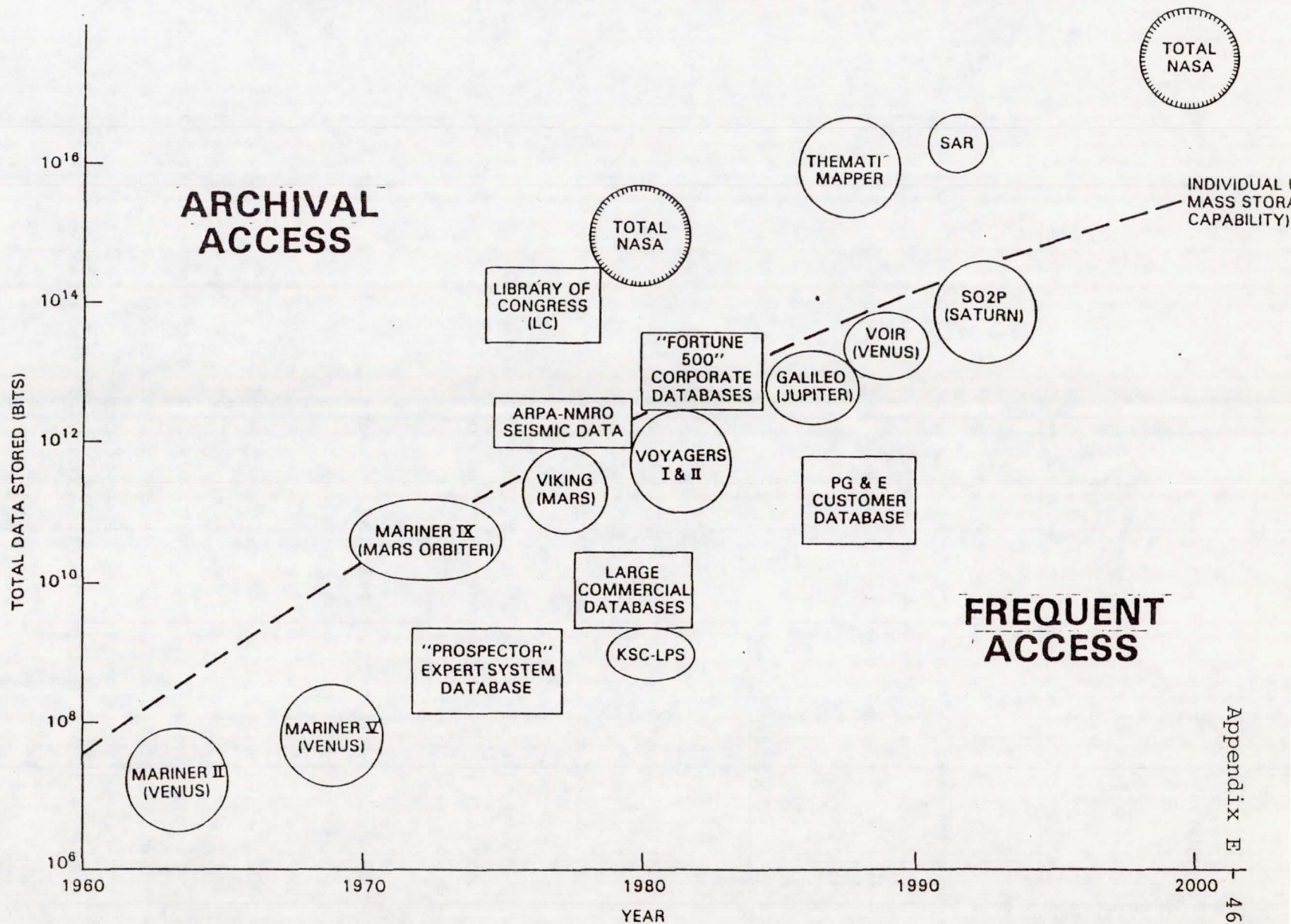


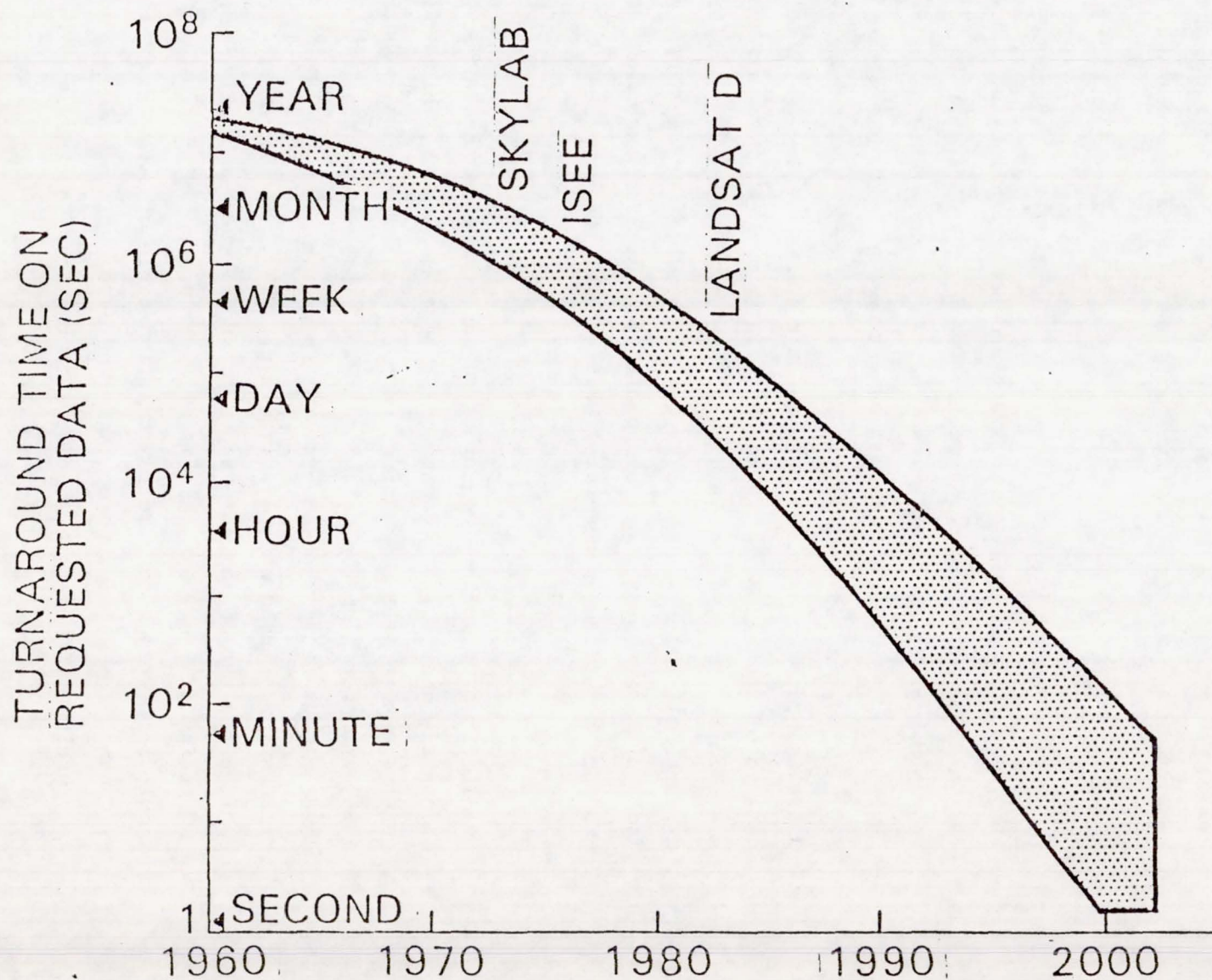
Figure E-13.

First, consider the bandwidth problem from the Agency-wide viewpoint (see Figure E-13). As of 1980, large commercial bibliographic databases require 1-10 Gb (10^9 - 10^{10} bits) of storage, and large insurance corporations have policy databases in the 10^{12} bit range. These data are meant to be searched often and exhaustively. At the other extreme, the total information stored by NASA, the Library of Congress, and other large governmental entities is about 10^{11} - 10^{14} . These archival data are never searched en bloc, and searches and retrievals are expected to be in limited domains or are understood to be mere samplings.

A reasonable dividing line between "archival" and "frequent-access" information is provided by the mass storage capability of individual users. Any database which exceeds in size the ability of a user to store it cannot be searched rapidly and exhaustively and should be viewed as an archive. Currently, individual-user mass memory commercially available is roughly 1 Tb (10^{12} bits) (Eastlake et al., 1979). Coincidentally, this is the total NASA planetary mission data returned to date, and also the on-line storage capacity desired by the planners of the OPEN (1979) project. In addition, one terabit seems to be the maximum sized base that might reasonably be searched exhaustively (at least once) in a given analysis or applications project. Individual-user mass storage capability of well in excess of 10^{15} bits is expected to become commercially available by the year 2000 A.D. (Whitney, 1976).

Information must be consolidated in a compact, accessible form to be adequately controlled. For the next several decades, this basic technological requirement implies a need for improved optical storage techniques (Allan, 1975; Waterworth and Reid, 1976). RTOP's have already been proposed to develop high capacity data systems, in particular a 10^{12} - 10^{15} Optical Archival Mass Memories (Reinbolt, 1974; Bailey, 1977). Such memories are intended to replace the use of magnetic tapes presently used in archival storage facilities of large database centers (Wildmann, 1975). Data stored archivally should be recorded on a medium that is stable or permanent, has a low cost per bit, can be recalled without degrading data quality, can be duplicated easily, and requires little maintenance. The result of such a development activity would be to produce a random access, read-write memory system -- probably

FUTURE NASA TURNAROUND REQUIREMENTS FOR DATA/INFORMATION REQUESTS



using holographic techniques, sophisticated memory architecture and computer interfaces, and high-quality optical memory components. Further research should focus upon key relationships bounding the technology choices for large, archival, digital storage devices, and clarify the motivations for selecting the optical technology ultimately for a petabit-exabit (10^{15} - 10^{18} bits) level storage system such as the BYTERON concept (Heard, 1979).

Future NASA data/information accessibility bandwidth needs are then determined by the speed with which a project manager or scientist requires data to be accessible to him (see Figure E-14). If present turnaround times of days or weeks are sufficient, then the required capacity is only about 10 Gb/sec by the turn of the century. If, however, the data are to be available within hours or minutes of a request, the necessary peak bandwidth rises into the 0.1-10 Tb/sec range (see Figure E-15).

Second, one might consider the bandwidth problem from the standpoint of communications technology available by the year 2000 A.D.. The Air Force Space Laser Communications achieved 1 Gb/sec in prototype testing, and the system has an inherent growth capacity of 8-10 Gb (Ross et al., 1978). The possibility of digital transmission at Gb/sec rates using fiber optical links has stimulated the development of gigabit electronic circuits necessary for transmitters and receivers. Gigabit/sec shift registers, modulators and multiplexers, asynchronous transceivers, photomultipliers, and holographic digital recorders all have been developed and tested on an experimental basis (Hance et al., 1971; Andrew and Johnson, 1973; Gardanier and Ryan, 1973; Bardos et al., 1976; Heinen et al., 1976; Mause et al., 1976; Teague and Allen, 1977; Brandt et al., 1978; Green, 1978). One representative of a major aerospace corporation suggested during a personal interview that within 5-10 years 180 mb/sec/fiber optical fibers would be commercially available as a mature technology, and that 144-fiber cables were reasonable within the same time frame -- a commercial 26 Gb/sec capacity by the year 1990. Extrapolating an expected 30% per year increase in optical trunk capacity beyond the ten year time frame given above, by the end of the century individual 0.5 Tb cable systems may be available (Elion and Elion, 1978; Midwinter, 1979; Howes and Morgan, 1980).

ACCESSIBILITY BANDWIDTH NEEDS

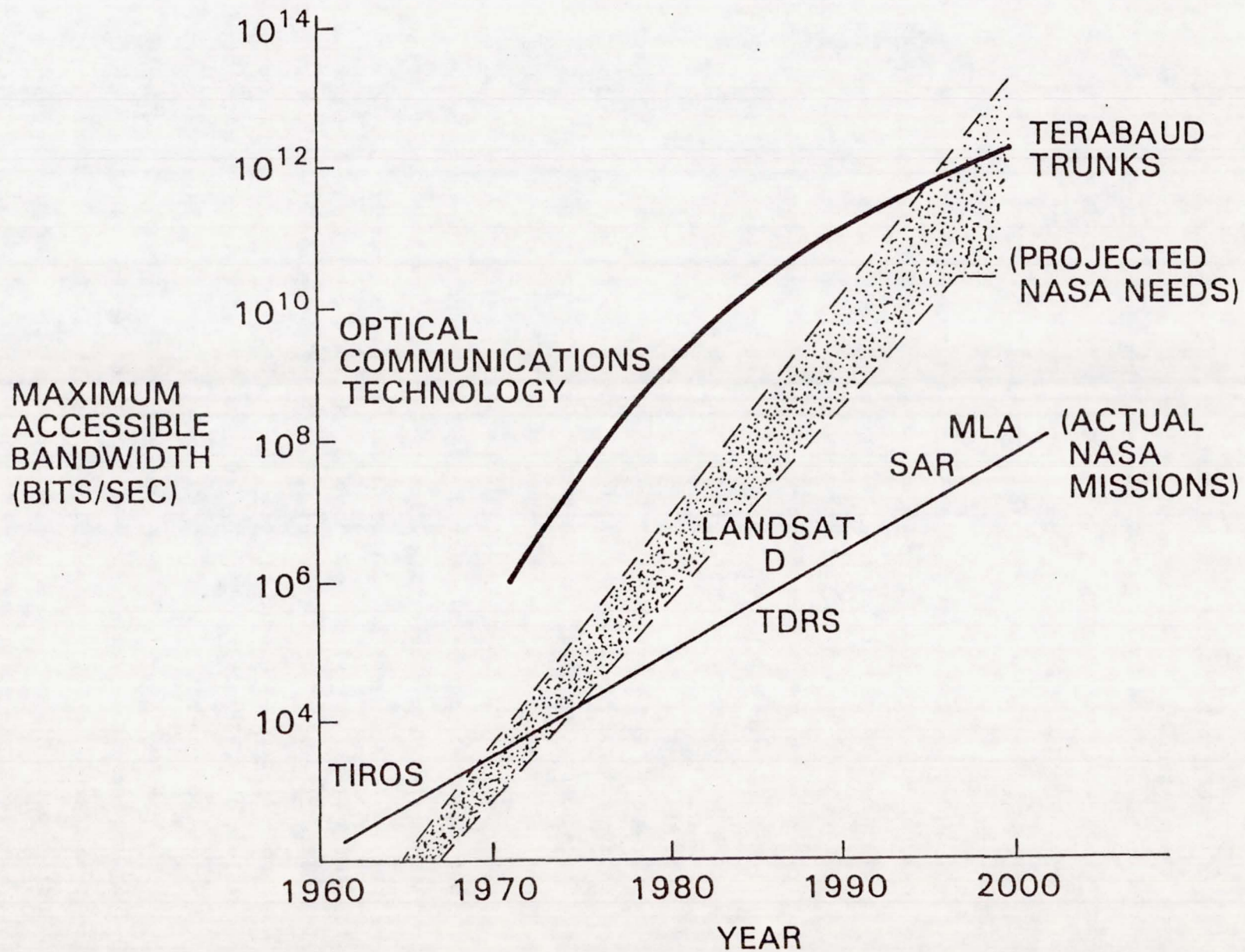


Figure E-15.

As a third approach, note that the total data-store accumulated by NASA since its inception amounts to about 10^{14} bits (CODMAC, 1981). Only about 1% of this tremendous resource has ever been analysed, and a far smaller fraction is readily accessible either to NASA personnel or to potential users from without. Over the years NASA has permitted intellectual control of this information slowly to slip from its grasp. Yet by the year 2000 A.D. it is estimated that the Agency will be in possession a staggering 10^{15} - 10^{17} bits of information. New techniques of processing are needed or else the analysis of such vast quantities of information is impossible and its accumulation becomes a pointless endeavor.

The technical difficulties of data processing in the 10^{12} - 10^{15} bit/sec range are just beginning to be seriously addressed (Grill and Weis, 1975), even though such figures are suggestive of the level of capabilities likely to be required by the turn of the century to support an information-intensive NASA (Whitney, 1976). A useful start in this direction, however, is the Massively Parallel Processor (MPP) project within the NEEDS program (Schaefer, 1980). MPP, scheduled for delivery in 1982, is an image-data processor intended to be "capable of accepting 10^{10} - 10^{12} bits/sec from several data channels, perform real-time operations on these bit streams, extract useful information, and reduce the bit stream required (for transmission to Earth) by at least three orders of magnitude."

But will such demand materialize by the year 2000? According to Posner (1979) of the Jet Propulsion Laboratory, ground fiber optics links will replace satellites for fixed services such as point-to-point trunking. By the start of the third millenium, guided waves will carry the wideband aggregated common carrier communications now handled by satellite links.

Already the Datacomputer (an operational general-purpose database system) is capable of handling data sets in excess of 10^{12} bits (Eastlake et al., 1979). Copious inexpensive storage is provided by the incorporation of an Ampex Tera-Bit Memory (TBM). The TBM was first installed in 1975 and is a video-tape-technology-based system with an average access time of 5 seconds (Wildmann, 1975). The immediate purpose of the project was to support the ARPA-NMRO seismic data activity by providing data storage and retrieval

services (CCA, 1976), but the system has since been used in other applications as well.

In this bandwidth-rich environment, NASA can significantly change its traditional manner of doing business. For example, a terabaud multi-cable trunk system connecting the Agency from coast to coast will give rise to an informationally-contiguous yet geographically-disjoint organization. Such a trunk provides approximately 50 Mb/sec capacity on a per-employee basis. In this environment video conferencing would become the rule rather than the exception. Tele-symposia "attended" by hundreds or even thousands of employees could take place within the Agency. Project Centers, Headquarters offices, key administrative personnel, and a variety of resource-sharing subgroups could be allocated (and independently operate) their own video channels within the closed-circuit NASA communication system -- indeed, a mere 0.1% of the available terabaud bandwidth is sufficient to support 1000 high-resolution (1 Mb/sect) video channels within the Agency.

At the personal level, the 50 Mb/sec allocation would permit local offices or even individuals to access up to 500,000 megabytes per day for searching or processing. The peak rate might almost never be used, but when necessary it will allow massive, full-text literature searches of entire "technical libraries" in hours, a thousand "books" in minutes, and full sets of specific "journals" in seconds. These three measures appear within quotation marks because by their current connotations they are misleading. By the year 2000 A.D., "books" and "journals" may no longer exist in their present form, having merged into large-scale knowledge bases. The "technical libraries" of tomorrow may be stored entirely on portable optical disks and could represent an amount of information equivalent to that presently stored in the entire Library of Congress. Clearly, massive amounts of computer science and technology will be required to retain intellectual control over such vast quantities of information.

There are, of course, many options which permit somewhat lower transmission rates. One alternative is to send the software to the data, not vice versa. This notion is valuable in many cases, especially in those instances where the raw data is stored in a straightforward sequential format

and the software is relatively simple. However, in the future it is likely that the data returned by NASA applications and exploratory satellites and vehicles will already be highly processed and may itself represent information of a fairly high order. The software for further processing will also have to be exquisitely sophisticated. In these cases it is unclear whether it is more cost-effective to transmit data to software or vice versa.

Other possibly negative factors associated with "traveling software" include:

- o Increased chance of error in software execution,
- o More difficult recovery from software failures,
- o Increased "middleman" costs, since software is inherently more fragile than data,
- o Increased potential problems associated with the use of less-compatible hardware, and
- o Removal of control from the local level with transfer to distant "network/database administrators."

E.4.2 Information Factories

An exciting new concept in computer science and automation which has received much recent attention is the notion of the self-replicating system or SRS (Long and Healy, 1980; Freitas, 1981). SRS is, in theory, the last physical manufacturing system humankind need ever build. Such a device would consist of a sufficient set of machines subsuming all basic manufacturing processes so that all higher-order processing machines can be built by the "properly instructed" system. Furthermore, SRS is open with respect to information, so it can be reprogrammed to produce any new product, or, perhaps more interestingly, any new machine capable of performing any new process that may be discovered in the future. This could also include processes which may already be known but are not yet being fully utilized.

The counterpart of the SRS in the realm of information processing, as distinct from the realm of mass-energy processing, is the Information Factory. Such systems would reside at the nodes of very-wide-bandwidth network trunks

INFORMATION FACTORIES

- HOSTS AT NODES OF WIDE BAND GENERAL RESOURCE SHARING NETWORK
- PRODUCES NEW INFORMATION INTERACTIVELY
- AUTONOMOUSLY GENERATES INFORMATION ON THE USER'S BEHALF
- RECURSIVELY IMPROVES ITSELF BY SELF-EDUCATION

and would permit users to access any information, any software package, any computer system, or any network which is publically available anywhere in the world. An Information Factory could interactively produce new information, or, ultimately, autonomously generate that information on the user's behalf. The total result is a comprehensive computer information consultant, capable of disseminating useful information and of recursively improving its own performance by self-education with or without direct human assistance (see Figure E-16).

The preceding description is, of course, the grand view. In the nearer term, the Information Factory concept should be applied to more restricted domains so that it might be more readily implemented. A major portion of NASA's organizational mission is the acquisition and processing of vast amounts of scientific, research, and operational data. Most of this is not accessible to the majority of Agency personnel, who often are surprised to learn that certain information relevant to their specific interests exists at another Center. Further, the decentralized NASA organizational structure makes coordination and Agency-wide planning virtually impossible without the availability of a good system for information dissemination and feedback. At present, Agency policies are too often motivated by crisis management considerations, which lead to optimization at the local level but suboptimization of resources Agency-wide. To regain intellectual control of its operations and information channels, NASA should take a leadership role in the development of technologies resulting in the ultimate goal of recursive information processing.

When fully implemented, the Information Factory would be interrogated either by keyboard or (ultimately) by natural language. Users would request information relevant to some problem they are working on. The questions may have ambiguous meanings, incomplete referents, or contain data of only questionable or partial validity. The goal of the system is, first, to discern what is the user actually wants, exactly what leads he can provide to assist the search, and precisely how confident he is of each of the data he has given to the machine. This accomplished, the system next sets up a search tree structure with alternative paths to intermediate search nodes, graded according to reliability and speed of response, availability, cost, and

INFORMATION FACTORIES

DEVELOPMENTAL PHASE

- I. RESPONSIVE DATABASES
- II. INTERACTIVE QUESTION — ANSWER
 - FUZZY REQUESTS
 - RELATIONAL DATA STRUCTURE
- III. INTELLIGENT PROBLEM SOLVING
 - EXPERT SYSTEMS
 - INTERACTIVE COMPUTING
- IV. RECURSIVE INFORMATION MANUFACTURING
 - INFORMATION SCIENCE EXPERT SYSTEM
 - SEMI-AUTONOMOUS SELF-IMPROVEMENT

Figure E-17.

overall probability of success. The system then implements the search, eventually distributing the final results, positive or negative, in any form convenient for the user.

Implementation by NASA (or any other organization) should proceed logically through four phases (see Figure E-17), each successive phase requiring somewhat more sophisticated machine intelligence, automation, and computer science capabilities than the previous level. These phases are: (1) Responsive Databases, (2) Interactive Question-Answering, (3) Intelligent Problem-Solving, and (4) Recursive Information Manufacturing.

Recursive Information Manufacturing

Information Factories would be difficult, though hardly impossible, to partially implement using present-day computer systems and techniques. Much of the decision tree architecture has already been worked out in chess-playing computer games and various "intelligent" expert systems developed at SRI International and elsewhere (Hayes-Roth et al., 1980; Buchanan, 1981). Content extraction programs are now being developed which are capable of abstracting printed matter, and a few existing systems already do a credible job in limited domains (Schank, 1976; Lebowitz, 1981). Reasonably general-purpose text abstractor are just beyond state-of-the-art, and may be available within 5-10 years. Fully relational databases should enter state-of-the-art within the next 5 years (Salton, 1979), adequate natural language capacity perhaps within 5-10 years, and sufficient multilinguistic and intertranslational capability within perhaps 15 years. The proposed Information Factory capabilities would not be beyond the capacity of an information specialist such as a librarian, a consultant, or a competent research scientist (particularly in his own field of expertise). Indeed, the search process closely mirrors the steps which a human expert might take to locate the desired information. Several dozen "expert systems" have been implemented in the last 10-15 years which involve one or more specialists in a particular field such as geology, topology, or biochemistry. In each case a programmer works with the specialist to identify his actual operational methodology for problem-solving and attempts to model it on a computer. There is no reason why the same thing could not be done with a specialist in

information science.

Indeed, some work has recently been done to automate more fully the now-expensive process of expert system production. EMYCIN (Van Melle, 1980) is one such "software tool" that helps a person design and build a MYCIN-like expert system. EMYCIN assumes that production rules are an appropriate representation framework for a person's new knowledge base and that a backward-chaining, or goal-directed, interpreter is an appropriate inference mechanism. If a new problem can be set up as a problem of gathering evidence for and against alternative hypotheses that define subgoals for ultimately satisfying the major goal, then EMYCIN is likely to provide some help in constructing an initial prototype expert system to solve the problem. Work is progressing rapidly in this field (Buchanan, 1981).

An expert system in information science would be qualitatively different from any other kind of expert system, just as tool-making tools are quite different from thing-making tools. In this instance, the former has the theoretical potential for self-replication and growth, and ultimately for dynamic self-improvement. On the other hand, the latter is static and can never produce any product other than that for which it was originally programmed. Information Factories could retrieve all available relevant information needed to solve a problem. This, of course, includes information about how to solve problems. With access to any other expert system which is developed, plus its "universal computational" (Turning machine) capability, recursive processing systems should be able to analyse problems and recommend solutions as well as any human specialist who has been properly modeled by a computer expert system. Augmentation or ramification of computational capability represents one important class of problems which, if solved with respect to a particular Information Factory module, could lead to increased performance and the possibility of yet further augmentation during successive recursions of the same process.

Attempts have already been made to automate the processes of human hypothesis-generation and discovery, and although progress has been slow some strides have been made (Gettys et al., 1979; Hayes-Roth et al., 1980). For example, BACON is a production system that discovers empirical laws by

incorporating some general heuristics that can lead to discovery in a number of domains (Langley, 1981). The main heuristics detect constancies and trends in data, and lead to the formulation of hypotheses and a definition of theoretical terms. BACON has the ability to carry out and relate multiple experiments, collapse hypotheses with identical conditions, ignore differences to let similar concepts be treated as equal, and to discover and ignore irrelevant variables. The program has demonstrated its efficacy by rediscovering versions of the ideal gas law, Kepler's third law of planetary motion, Ohm's law, and Galileo's laws for the pendulum and constant acceleration.

It is clear that an Information Factory will have the potential capability for self-improvement, either with or without direct human assistance. To give an example, it is likely that such a complex system will require a number of internal programmatic indicators to continuously monitor levels of performance. This might include response times, percent of user requests successfully completed, mean time to completion, and so forth. Now suppose that a "demon" program is running continuously in the background, performing its computing only in off-peak hours when the system is not being fully utilized and spare time is available. The demon looks at the data coming from the system monitors, compares them to norms, and recognizes any significant deviations. Ultimately, the demon could also select various functions which were not being performed as efficiently as most others and could use some improvement.

Having identified some problem or set of problems, the demon then queries the Factory controller directly, asking it to generate any materials which might possibly have a bearing on the solution of the problem(s). In the simplest case, these materials would be reported to a human supervisor to assist him in upgrading relevant software components. In a more sophisticated version, the system might abstract or perform a partial analysis of the materials collected, and then propose major components of a self-upgrade plan to the human supervisor, together with the supporting documentation, for his approval and implementation. In the most sophisticated version of a self-learning system, the proposal for an improvement would be planned and executed entirely autonomously without a human in the loop.

Of course, if additional hardware is required the system is powerless until some human being plugs in that equipment. Nevertheless, Information Factories are information self-sufficient, in the sense that they may have access to all literature and have direct call-up capability to other computers and to specific people if they need to ask questions in an interactive mode. The most sophisticated versions should also be able to perform information abstraction, general problem-solving, and planning without human intervention -- in addition to their "intelligent" information acquisition capabilities. Thus they can, in their fully implemented version, produce any information which can be acquired or derived from current human knowledge and which is not logically, formally, or sociopolitically impossible to obtain. However, the Information Factories will require matter-energy inputs such as power, new components, maintenance, and physical repair services. It is interesting that recursive information systems are the converse of the recent concept of Self-Replicating Systems (SRS). The latter are expected to be largely matter-energy self-sufficient and could produce virtually any material output which human beings (or their artifacts) can describe, but require information inputs to deal with novel situations or to be taught how to manufacture new kinds of tools or output product types. In other words, Information Factories are "general product factories" for information, whereas the SRS is a "general product factory" for physical materials.

Technical and Social Barriers to Implementation

The major technical assumption in the concept of Information Factories is that a variety of expert systems, software packages, networks, and interactive databases will come on-line for the general public in the next decade or two. Judging from the recent explosive growth of such systems, it seems likely that a decade from now there may be literally thousands of such systems to choose from. This seems all the more reason to automate the selection and interfacing aspects of dealing with these myriad informational resources.

Another major technical requirement for such a system is an ultra-high bandwidth, national network capability. Traffic calculations presented earlier have revealed the extent of the problem. Current data channels are inadequate to meet such an enormous demand. A nationwide optical fiber

terabaud-capacity network may be the only feasible way to implement a full-capacity Information Factory system.

Social barriers to implementation are likely to prove more difficult than the technological hurdles. To take a simple example, owners of specific databases or expert systems may choose to restrict access to a small community of private users. This will be especially the case if the software in question has a proprietary or research character.

Another example involves the human-machine interface and public access to particular individuals. It is somewhat naive to expect that all professionals will share their expertise, free of charge, at the ring of a telephone. To the extent professionals may charge standard fees for their services, this objection may be overcome. Many people may resent being telephoned and interrogated by an inquisitive software package; others will demand that their names be removed from all relevant public directories. At present, for instance, most well-known authors do not make public their phone numbers or addresses. Even though this information is publically available (in Who's Who directories, voting registration listings, and the like), it is sufficiently difficult to obtain to deter all but the most determined seekers. Should Information Factory technology become widely available, all these sources could be electronically searched, destroying the privacy of many individuals.

From a broader philosophical perspective, knowledge is power -- and a "Phase IV" system is the ultimate knowledge factory. Whoever controls recursive information manufacturing technology has the potential for comprehending, even manipulating, large, complex socioeconomic or political systems for personal gain or for mischievous purposes.

E.4.3 An Intelligent Earth-Sensing Information System

In a previous Summer Study report (Long and Healy, 1980; Freitas, 1981), the Terrestrial Applications Team recommended the development of an Intelligent Earth-Sensing System (IESIS) consisting of cooperative autonomous satellites in low Earth orbit communicating with a ground processing facility. This mission was proposed as a partial solution to the increasingly critical

bandwidth and data storage problem. These technical issues aside, remote sensing of the Earth is a research area of rapidly growing importance, where NASA leads in technological expertise. The following is a brief outline of the major concepts described in the above reports, with a focus here on the computer science issues involved.

The heart of the IESIS system is a world model: A compact representation of persistent or predictable spatial and temporal characteristics of the Earth, its lands, oceans, and atmosphere, together with algorithms for the use of the representation.

The system operates autonomously to:

- o Routinely acquire data on board,
- o Process data through comparison with the world model,
- o Transmit abstracted data generally, and more detailed information if an anomaly is detected,
- o Update the world model, and
- o Archive information.

A user can directly address the system interactively using natural language to:

- o Request specific information,
- o Obtain cost estimates for requests,
- o Request algorithms for on-board processing,
- o Supply new algorithms, and
- o Cancel or modify requests.

The world model consists of two separate components. The first is the state component (database) which defines the state of the world to a predetermined level of accuracy and completeness. Second is the theory component, which consists of a set of interacting expert knowledge systems which allow:

- o Abstraction of useful information,

- o Large-scale data reduction, and
- o Learning by experience.

This Earth-sensing system is more completely described in the references cited earlier, and also in the 1980 Summer Study Team paper (Fay et al., 1981).

The need for such an intelligent system comes from several points. With the world population expected to increase by more than 50% by the year 2000, there is an obvious need for prudent terrestrial resource management. Remote sensing is a major tool in resource management. Another major factor is cost considerations. Presently NASA spends on the order of \$700 million per year for data handling (OAST, 1978). Without more efficient data handling techniques, the cost of obtaining information in a timely manner will grow out of proportion to its value. Furthermore, if NASA expects to move toward deep-space exploration with intelligent spacecraft, world model development and on-board data processing are requisites. Sensing the Earth autonomously thus also becomes a pilot project for many of these ultimate endeavors.

E.5 References

Adler, E., Haupt, H. and Zschunke, W., "A 34 Mbit/s Optical Field Trial System," Proc. 4th Europ. Conf. Optical Communication, Genova, 12-15 September 1978, pp. 588-596.

Allan, R., "Semiconductor Memories," IEEE Spectrum, Vol. 12, August 1975, pp. 40-45.

Andrew, J.D. and Johnson, T.J., "High Speed Bit Synchronizer for Mode-Locked Laser Communications," in NAECON 73, Proceedings of the National Aerospace Electronics Conference, Dayton, Ohio, 14-16 May 1973, Institute of Electrical and Electronics Engineers, Inc., New York, 1973, pp. 151-158.

Baack, C., Elze, G., Fussgaenger, K. and Matt, H.J., "The HHI Broad-Band Fiber-Optic Communications System at Various European PCM Hierarchy Levels -- Status of Development and Preliminary Experimental Results," in ICC '79, International Conference on Communications, Boston, Massachusetts, 10-14 June

1979, Conference Record, Volume 2, Institute of Electrical and Electronics Engineers, Inc., Piscataway, New Jersey, 1979, pp. 28.6.1-28.6.5.

Bailey, G.A., "High Capacity Data Systems," NASA-Marshall Space Flight Center, Huntsville, Alabama, RTOP (77W70316, 506-20-13), 1977.

Bansal, Sam, "Get More Out of a Computer -- Link It to a Management and Control Network," Control Engineering, May 1981, pp. 159-167.

Bardos, A.W., Moore, G.S., Nelson, R.H., Ralston, L.M. and Shareck, M.W., Wideband Holographic Digital Recording and Reproduction, Final Technical Report, January 1973 - November 1976, Electronic Systems Division, Harris Corporation, Melbourne, Florida.

Barstow, D., "An Experiment in Knowledge-Based Automatic Programming," Artificial Intelligence, Vol. 12, 1979, pp. 73-119.

Bell System Technical Journal, Vol. 57, 1978, pp. 1717-1895. (Whole issue, No.6)

Berry, R.W., Brael, D.J. and Ravenscroft, I.A., "Optical Fiber System Trials at 8 Mbit/s and 140 Mbit/s," IEEE Trans. Communications, Vol. COM-26, 1978, pp. 1020-1027.

Boraiko, Allen A., "Fiber Optics: Harnessing Light by a Thread," National Geographic, Vol. 156, October 1979, pp. 516-534.

Brandt, G.B., Gottlieb, M. and Weinert, R.W., "Giga-Hertz Modulators Using Bulk Acousto-Optic Interactions in Thin Film Waveguides," in AGARD: Optical Fibres, Integrated Optics and Their Military Applications, Westinghouse Research Laboratories, Pittsburgh, PA, 1978.

Branscomb, L.M., "Computer Communications in the Eighties -- Time to Put it All Together," Computer Networks, Vol. 5, 1981, pp. 3-8.

Buchanan, Bruce G., "Research on Expert Systems," Stanford University,

Computer Science Department, Heuristic Programming Project, Report No. HPP-81-1, February 1981.

Buchanan, B.G. and Feigenbaum, E.S., "DENDRAL and Meta-DENDRAL: Their Applications Dimension," Artificial Intelligence, Vol. 11, 1978, pp. 5-24.

Buchanan, B.G., Mitchell, T., Smith, R.G. and Johnson, C.R., Jr., "Models of Learning Systems," in J. Belzer, ed., Encyclopedia of Computer Science and Technology, Volume 11, Marcel Dekker, Inc., New York, 1978.

Cahn, D.F. and Herr, J.J., "Automatic Database Mapping and Translation Methods," paper presented at the 4th Conf. of Cybernetics and Systems, Amsterdam, Netherlands, 21 August 1978.

Carson, A.N., "Large Capacity, High-Speed Holographic Memory," Proceedings of the Tutorial Seminar on Coherent Optics in Mapping, Rochester, NY, 1974, pp. 157-160.

CCA (Computer Corporation of America), "Datacomputer Support of Seismic Data Activity," CCA Quarterly Technical Report, 1 August 1975 - 31 October 1975, Computer Corporation of America, Cambridge, MA, 1976.

Clancy, W., "Tutoring Rules for Guiding a Case Method Dialogue," International Journal of Man-Machine Studies, Vol. 11, 1979, pp. 25-49.

CODMAC, Data Management and Computation: Issues and Recommendations, Space Science Board, Committee on Data Management and Computation, Draft 10, May 1981. National Academy of Sciences, National Research Council. A.G.W. Cameron, Chairman.

Dalke, E.A., "Information Management Systems," Houston, NASA RTOP, 1978.

DaSilvamartins, I., An XPL/ALGOL Translator, M.S. Thesis, Report No. INPE-1304-TPT/096, Instituto de Pesquisas Espaciais, Sao Jose dos Campos, Brazil, 1978.

Davis, R., "Applications of Meta-Level Knowledge to the Construction, Maintenance and Use of Large Knowledge Bases." Ph.D. Dissertation, Stanford University Computer Science Department, STAN-CS-76-564, 1976.

DeKleer, J., "Causal and Teleological Reasoning in Circuit Recognition," Ph.D. Dissertation, MIT Department of Electrical Engineering and Computer Science, MIT AI Lab Report TR-529, 1979.

Dickey, F.J., Translations Between Programming Languages, Ph.D. Thesis, Ohio State University (Univ. Microfilms 7922475), 1979.

Duda, R.O., Gaschnig, J., Hart, P.E., Konolige, K., Reboh, R., Barrett, P., and Slocum, J., Development of the PROSPECTOR Consultation System for Mineral Exploration, Final Report, SRI Projects 5821 and 6415, SRI International, Menlo Park, California, 1978.

Dunning, B.B., "Self-Learning Data-Base for Automated Fault Localization," in AUTOTESTCON '79, Proceedings of the International Automatic Testing Conference, Minneapolis, Minnesota, 19-21 September 1979, Institute of Electrical and Electronics Engineers, Inc., New York, 1979, pp. 155-157.

Earman and Lesser, "Hearsay II: Tutorial Introduction and Retrospective View."

Eastlake, D.E., III, Maltzman, M. and Sattley, J.Z., "Datacomputer and SIP Operations," CCA Final Technical Report for 1978, Computer Corporation of America, Cambridge, MA, 1979.

Elion, G. and Elion, H., Fiber Optics and Communication Systems, Dekker, New York, 1978.

English, W., "Study on Intersatellite Laser Communication Links," Battelle Institute, Frankfurt am Main (West Germany), Report No. BF-R-63.749-5 ESA-CR(p)-1259, 1980.

Erman, L.D. and Lesser, V.R., "A Multi-Level Organization for Problem Solving Many, Diverse, Cooperating Sources of Knowledge -- Artificial Intelligence,"

Department of Computer Science, Carnegie-Mellon University, 1976.

Faria, A.D., Simoni, P.O., "Using Representative Knowledge to Guide Decision Making in Automatic Diagnostic systems," paper presented at the 6th Congress, Brasileiro de Engenharia Biomedica, 24-27 October 1979.

Fay, T.H., Nazemetz, J., Sanford, D., Taneja, V., and Walsh, P., "An Intelligent Earth Sensing Information System," Photogrammetric Engineering and Remote Sensing, 1981, in press.

Freitas, R.A., Jr., ed., Advanced Automation for Space Missions, Final Report of the 1980 NASA/ASEE Summer Study on the Feasibility of Using Machine Intelligence in Space Applications, University of Santa Clara, Santa Clara, CA, 15 May 1981. 1200 pp.

Fry, J.P. and Merten, A.G., Toward the Development of a Data Translation Methodology and Selection of Target Database Structures, Final Report, 1 January 1976 - 28 February 1977, Department of Industrial and Operations Engineering, Michigan University, Ann Arbor, 1977.

Fussgaenger, K., Matt, H.J., Nobis, R., Saniter, J., Sauermann, K.D. and Schaeffner, H.P., "Test System for Wideband Communication via Optical Channels," in Laser 79 Opto-Electronics, Proceedings of the Fourth Conference, Munich, West Germany, 2-6 July 1979, IPC Science and Technology Press, Ltd., Guildford, Surrey, England, 1979, pp. 142-147. In German.

Gardanier, L. and Ryan, C.R., "Gigabit/Second Asynchronous Multiplexer and Demultiplexer," in National Telecommunications Conference, Atlanta, Georgia, 26-28 November 1973, Conference Record, Volume 2, Institute of Electrical and Electronics Engineers, Inc., New York, 1973, pp. 30G-1 - 30G-4.

Gettys, C., Mehle, T., Baca, S., Fisher, S. and Manning, C., A Memory Retrieval Aid for Hypothesis Generation, Progress Report, August 1978 - August 1979, Report No. TR-27-7-79, Decision Processes Laboratory, Oklahoma University, Norman, Oklahoma, 1979.

Glorioso, Robert M. and Osorio, Fernando C. Colon, Engineering Intelligent Systems, Digital Press, Bedford, MA, 1980. Chapter 9, "Stochastic Automata Models in Computer and Communication Networks."

Goldman, G., "Holographic Data Storage Using Spatial Frequency Multiplexing," in International Optical Computing Conference, Washington, D. C., 1975, pp. 98-104.

Green, S.I., "Gigabit Detectors for Visible Space Laser Communications," in ITC/USA/'78, Proceedings of the International Telemetering Conference, Los Angeles, California, 14-16 November 1978, Instrument Society of America, Pittsburgh, PA, 1978, pp. 333-337.

Greenblatt, R.B., Eastlake, D. and Crocker, S., "The Greenblatt Chess Program," Proceedings of the 1967 Joint Computer Conference 30, 1967, pp. 801-810.

Grill, W. and Weis, O., "Excitation of Coherent and Incoherent Terahertz Phonon Pulses in Quartz Using Infrared Laser Radiation," Physical Review Letters, Vol. 35, 1 September 1975, pp. 588-591.

Guedez, S., "Information Storage on Board Satellites and Space Vehicles," Le Recherche Spatiale, Vol. 13, 1974, pp. 5-12.

Haggerty, James J., ed., Spinoff 1981: An Annual Report, GPO, Washington, D.C., April 1981.

Haller, U., Herold, W. and Ohnsorge, H., "Problems Arising in the Development of Optical Communication Systems," Appl. Phys., Vol. 17, 1978, pp. 115-122.

Halliday, T.C. and Briss, E.A., "The Evaluation and Systems Analysis of the SYSTRAN Machine Translation System," Final Technical Report, November 1974 -- August 1976, Battelle Columbus Laboratories, Ohio, 1977.

Hance, H.V., Chow, K.K., Ohlmann, R.C., Peterson D.G. and Ward, R.B., "Modulation/Demodulation Techniques for an Optical One-Gigabit/sec

Intersatellite Data Transmission System," in International Telemetry Conference, Washington, D.C., 27-29 September 1971, Proceedings, International Foundation for Telemetry, Woodland Hills, CA, 1971, pp. 267-271.

Hara, E. H., "Conceptual Design of a Switched Television Distribution System using Optical-Fiber Waveguides," IEEE Trans. Cable Television, Vol. CATV-2, 1977, pp. 120-130.

Hayes-Roth, F., Klahr, P. and Mostow, D.J., "Knowledge Acquisition, Knowledge Programming, and Knowledge Refinement," RAND Corporation, Santa Monica, CA, 1980.

Heard, H. G., "Wideband Optical Storage," in Holography: A View of the Future, Proceedings of the Joint Symposium of the American Institute of Aeronautics and Astronautics and the World Future Society, Los Angeles, California, 20 January 1979, AIAA Inc., 1979, pp. 48-58.

Heinen, J., Huber, W. and Harth, W., "Light-Emitting Diodes with a Modulation Bandwidth of more than 1 GHz," Electronics Letters, Vol. 12, 14 October 1976, pp. 553-554.

Hermer, C., "A Forecast of the Future of Computation," in American Institute of Aeronautics and Astronautics, 13th Annual Meeting and Technical Display, incorporating the Forum on the Future of Air Transportation, Washington, D.C., 1977.

Hershman, R., Kelly, R.T. and Miller, H.G., User Performance with a Natural Language Query System for Command Control, NPRDC-TR-79-7, NAVY Personnel Research and Development Center, San Diego, California, 1979.

Hoffman, Lance J., "The Application of Networks in Basic Computer Science and Information Science Research," Information Processing and Management, Vol. 15, 1979, pp. 269-280.

Howes, M.J. and Morgan, D.V., eds., Optical Fibre Communications, John Wiley and Sons Ltd., Chichester, England, 1980, pp. 1-26.

JPL (Jet Propulsion Laboratory), Structured Programming-to-FORTRAN Translator: SFTRAN, JPL/Cal Tech Report No. NPO-13602 (NASA 75M10102).

Ke, J.S., Database Modeling Techniques with Application to Fuzzy Query Translation, Ph.D. Thesis, Illinois University at Chicago Circle, Chicago, Illinois (Univ. Microfilms 7824357), 1979.

Kennedy, C.J., "The Growth Potential of Nd:YAG Space Laser Communications," paper delivered at the Spring Meeting of the Optical Society of America, Washington, D.C., 21-25 April 1974.

Kornhuber, H., "Cerebral Cortex Cerebellum and Basal Ganglia: An Introduction to their Motor Functions," in The Neuro Sciences: Third Study Program, pp. 267-280.

Kozma, A., "Holographic Storage and Recording of Computer Data," in Laser Recording and Information Handling Technology, Proceedings of the Seminar, San Diego, 1974, pp. 77-83.

Kuo-Petravic, G., Petravic, M. and Roberts, K.V., A User's Guide to the Translator of Symbolic ALGOL 1 to Symbolic ALGOL 2 (TRANSTAG), Department of Engineering Science, Oxford University, England, 1972.

Langley, Pat, "Data-Driven Discovery of Physical Laws," Cognitive Science, Vol. 5, 1981, pp. 31-54.

Lebowitz, M., Generalization and Memory in an Integrated Understanding System, Report No. RR-186, Department of Computer Science, Yale University, New Haven, Connecticut, 1981.

Liertz, H., Goldmann, H. and Schicketanz, D., "Laying and Properties of an Optical Wave Guide Test Link in Berlin," Proc. 4th Europ. Conf. Optical Communication, Genova, 12-15 September 1978, pp. 583-587.

Lindsay, R., Buchanan, B.G., Feigenbaum, E.A. and Lederberg, J., "Applications of Artificial Intelligence for Organic Chemistry: The DENDRAL Project,"

McGraw-Hill, New York, 1980.

Long, James E. and Healy, Timothy J., Advanced Automation for Space Missions, Technical Summary, 15 September 1980. NASA CR-163827.

Martin, M.J., EDALG -- A Computer Program to Translate Elliott 4100 ALGOL into ICL 1900 ALGOL, Report No. RAE-TR-70230 RAD-1048, Volume 1 (General Description and User's Guide), Royal Aircraft Establishment, Farnborough, England, 1971.

Mause, K., Hesse, E. and Schlachetzki, A., "Shift Register with Gunn Devices for Multiplexing Techniques in the Gigabit-per-Second Range," IEEE Journal on Solid-State and Electron Devices, Vol. 1, September 1976, pp. 17-23.

Melton, R.A., Structured Programming Translators, Volume 1, Final Technical Report, May 1975 - January 1976, General Research Corporation, Santa Barbara, California, 1977.

Michels, D.L., "A Concise Extensible Metalanguage for Translator Implementation," paper presented at the USA-Japan Computer Conference, San Francisco, California, 1978.

Midwinter, John E., Optical Fibers of Transmission, John Wiley and Sons, New York, 1979.

Mitchell, T.M., "Version Spaces: An Approach to Rule Revision During Rule Induction," Proceedings IJCAI-77, 1977, pp. 305-310.

Moore, R. C., Mechanical Intelligence: Research and Applications, Final Report, 1 October 1977 -- 30 September 1978, Artificial Intelligence Center, SRI International, Menlo Park, California, 1979.

Moses, J., "Symbolic Integration: The Stormy Decade," Communications ACM, Vol. 8, 1971, pp. 548-560.

Murray, W.J., "Economic Planning and System Parameter Objectives for Optical

Fibre Transmission Systems in the British Post Office," Proc. 4th Europ. Conf. Optical Communication, Genova, 12-15 September 1978, pp. 502-509.

Myers, Ware, "Computer Graphics: The Human Interface," Computer, Vol. 13, June 1980, pp. 45-54.

Nakahara, T., Kumamanu, H. and Takeuchi, S., "An Optical Fiber Video System," IEEE Trans. Communications, Vol. COM-26, 1978, pp. 955-961.

National Bureau of Standards, "Message Handling Facilities: Models and Services," Version 1, NBS, Gaithersburg, MD, 1981, pp. 26.

National Bureau of Standards, "Message Handling Facilities, Models and Services," Decided at the Third preparatory Meeting on Message Handling Facilities, The Hague, March, 1981, 26 + vii pages.

Nilson, N.J., A Production System for Automatic Deduction, Report No. SU-STAN-CS-77-618, Department of Computer Science, Stanford University, Stanford, California, 1977.

Nonweiler, T.R.F., A Systematic Translation of the IMP Language into PL/I, Edinburgh Regional Computing Centre, Edinburgh University, Scotland, 1974.

OAST, "Program Plan for the NASA End-to-End Data System Programs, Phase 2," November 1978.

OPEN (Origin of Plasmas in the Earth's Neighborhood), Final Report of the Science Definition Working Group, NASA-Goddard Space Flight Center, Greenbelt, Maryland, April 1979.

Osterweil, L.J. and Myers, E.W., Jr., FLAT: A FORTRAN Language Augmentation Tool, Department of Computer Science, Colorado University at Boulder, 1981.

Pan, Anthony, "Integrating Voice and Data Traffic in a Broadcast Network Using Random Access Scheme," in Evolutions in Computer Networks, Proceedings of the Fourth International Conference on Computer Communications, North Holland

Publishing Co., Amsterdam, 1978, pp. 545.

Pople, H. E., "The Formation of Composite Hypotheses in Diagnostic Problem Solving -- An Exercise in Synthetic Reasoning," Proceedings IJCAI-77, 1977, pp. 1030-1037.

Posner, Edward C., "Information and Communication in the Third Millenium," IEEE Communications Magazine, Vol. 17, January 1979, pp. 8-15.

Pribram, Karl, The Holographic Hypothesis of Memory Structure in Brain Function and Perception, 1978.

Reinbolt, E.J., "Optical Mass Memory," NASA-Marshall Space Flight Center, Huntsville, Alabama, RTOP (74W70357, 502-23-31), 1974.

Reiter, R., An Approach to Deductive Question-Answering, Report No. BBN-3649, Bolt, Beranek, and Newman Inc., Cambridge, Massachusetts, 1977.

Restak, Richard M., The Brain: The Last Frontier, Doubleday, NY., 1979.

Roland, J.R. and Whited, C.E., "Air Force Space Laser Communications," in Laser and Fiber Optics Communications, Proceedings of the Seminar, San Diego, California, 28-29 August 1978, Society of Photo-Optical Instrumentation Engineers, Bellingham, Washington, 1978, pp. 2-8.

Rosenberg, S., Understanding in Incomplete Worlds, Report No. AI-M-475, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, 1979.

Ross, M., Abernathy, J. and Wolf, J., "Space Laser Communications," in 7th Communications Satellite Systems Conference, San Diego, California, 24-27 April 1978, American Institute of Aeronautics and Astronautics, Inc., New York, 1978, pp. 448-456.

Ruell, H., "Holographic Storage of Information in Narrow Tracks," in Laser 75 Opto-Electronics, Proceedings of the Conference, Munich, 1975, pp. 241-247.

Russer, P., "Introduction to Optical Communications," in M. J. Howes, D. V. Morgan, eds., Optical Fibre Communications, John Wiley and Sons Ltd., Chichester, England, 1980, pp. 1-26.

Sacerdoti, E.D., Fikes, R.E., Hendrix, G.G., Morris, P. and Sagalowicz, D., Mechanical Intelligence: Research and Applications, Final Technical Report, 12 April 1976 - 9 October 1977, SRI International, Menlo Park, California, 1978.

Sagan, Carl, Chmn., Machine Intelligence and Robotics: Report of the NASA Study Group, Final Report, NASA, Washington, D.C., 1980.

Salton, Gerard, "SMART", in Jack Belzer, Albert G. Holzman, Allen Kent, eds., Encyclopedia of Computer Science and Technology, Volume 13, Marcel Dekker, Inc., New York, 1979, pp. 137-172.

Samuel, A.L., "Some Studies of Machine Learning Using the Game of Checkers," IBM Journal of Research and Development, Vol. 3, 1959, pp. 211-229.

Schaefer, David H., NASA End to End Data System (NEEDS) Massively Parallel Processor (MPP) Concept Document, prepared by Concept Development Section, NASA-Goddard Space Flight Center, Greenbelt, Maryland, 30 May 1980.

Schank, R.C., SAM: A Story Understander, Department of Computer Science, Yale University, New Haven, Connecticut, 1976.

Secretariat (USA), ISO (International Organization for Standardization), Reference Model of Open Systems Interconnection (Version 4, as of June 1979).

Shortliffe, E.H., Computer-Based Medical Consultations: MYCIN, American Elsevier, New York, 1976.

Shrobe, H.E., Dependency Directed Reasoning for Complex Program Understanding, Report No. AI-TR-503, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, 1980.

Smylie, Robert E., "NASA Activities and Plans," presented at the AIAA/NASA Symposium on Space Tracking and Data Systems, Washington, D.C., June 1981, pp. 10.

Stanten, S.F. and Flanders, J.H., The GOAL-to-HAL/S Translator Specification for Space Shuttle, Intermetrics, Inc., Cambridge, Massachusetts, 1974.

Stickel, M.E., Mechanical Theorem Proving and Artificial Intelligence, Ph.D. Thesis, Carnegie-Mellon University, Pittsburgh, Pennsylvania (Univ. Microfilms 78-12160), 1978.

Sussman, G.A., A Computer Model of Skill Acquisition, American Elsevier, New York, 1975.

Teague, J.R. and Allen, L.B., "One Gigabit Wideband Modulator for Space Laser Communication System," in High Speed Optical Techniques, Proceedings of the Seminar, San Diego, California, 26-27 August 1976, Society of Photo-Optical Instrumentation Engineers, Palos Verdes Estates, CA, 1977, pp. 142-150.

Van Bergeijk, D., "From ETC to ITC: The International Translations Centre," in AGARD: The Impact of Future Developments in Communications, Information Technology and National Policies on the Work of the Aerospace Information Specialist, 1978.

Van Melle, W., "A Domain Independent System that Aids in Constructing Knowledge Based Consultation Programs," Ph.D. Dissertation, Stanford University Computer Science Department, STAN-CS-80-820, 1980.

Von Braun, Werner, "Prospective Space Developments -- Blue Sky and Non-Blue Sky," in Astronautics and Aeronautics, Vol. 10, April 1972, pp. 26-35.

Waterman, D.A., "Generalization Learning Techniques for Automating the Learning of Heuristics," Artificial Intelligence, Vol. 1, 1970, pp. 121-170.

Waterworth, P. and Reid, D.C.J., "Holographic Data Storage," in The

Engineering Uses of Coherent Optics, Proceedings of the Conference, Glasgow, Scotland, 8-11 April 1975, Cambridge University Press, 1976, pp. 317-336.

Whitney, W.M., "Section IV. Processing and Storing Information," in A Forecast of Space Technology 1980-2000, NASA SP-387, 1976, pp. 3-78 - 3-126.

Wildmann, M., "Terabit Memory Systems -- A Design History," IEEE Proceedings, Vol. 63, August 1975, pp. 1160-1165.

Wilkins, D., "Using Patterns and Plans in Chess," Artificial Intelligence, Vol. 14, 1980, pp. 165-203.

Wilkins, D.E., Research on Parallelism in Problem-Solving Systems, Annual Technical Report, Artificial Intelligence Center, SRI International, Menlo Park, California, 1981.

Winston, P. H., "Learning and Reasoning by Analogy: The Details," MIT AI Lab Memo, No. 520, 1979.

Wood, R. J., Computer Aided Program Synthesis, Technical Report TR-861, Department of Computer Science, University of Maryland, College Park, Maryland, 1981.

Woods, W. A., Kaplan, R. M. and Nash-Webber, B., "The Lunar Sciences Natural Language Information System," Bolt Beranek and Newman, Cambridge, MA, June 1972.

Zadeh, L.A., "Theory of Fuzziness and Its Applications to Information Processing and Decision-Making," University of California, Berkeley, Electronics Research Laboratory, 1979.

Appendix F. An Executive Guide to the Computer Age

In scientific prognostication we have a condition analogous to a fact of archery -- the farther back you are able to draw your longbow, the farther ahead you can shoot.

-- Buckminster Fuller (1981)

The assumption behind this Appendix is that computing technologies have the potential to radically transform the way we live. Such a transformation is not inevitable, nor is it necessarily good. But the possibility of radical transformation has been explored by a variety of modern thinkers, including Herbert Simon, Seymour Papert, James Albus, Alvin Toffler, and Buckminster Fuller. The purpose here is to place this possibility into its proper historical perspective and to consider, in a general way, how one plans for it.

The first section considers the place of the "Information Age" in history, suggesting that it is the fourth major transformation in human cultural evolution. The next section develops a five-dimensional metaphor outlining certain basic factors which may be considered in a strategic plan for the use of computing technology. The final section discusses a specific aspect of that planning -- the relationship between computing and productivity -- and suggests that the transforming power of computing technology lies in the possibility of dramatically increasing productivity as intelligent computing becomes routine and reliable.

F.1 The Information Age in History

Close examination of a number of accounts of cultural evolution (White, 1959; Tatje and Naroll, 1970; Harris, 1977; Boulding, 1978) suggests that culture evolves through distinct phases involving global transformations of economic, organizational, and communication structure (see Figure F-1). The first major transition occurred perhaps 50,000 -- 100,000 years ago and converted isolated packs of intelligent apes into bands of hunting-gathering human beings. These people existed by extracting food from the environment;

THE INFORMATION AGE IN HISTORY

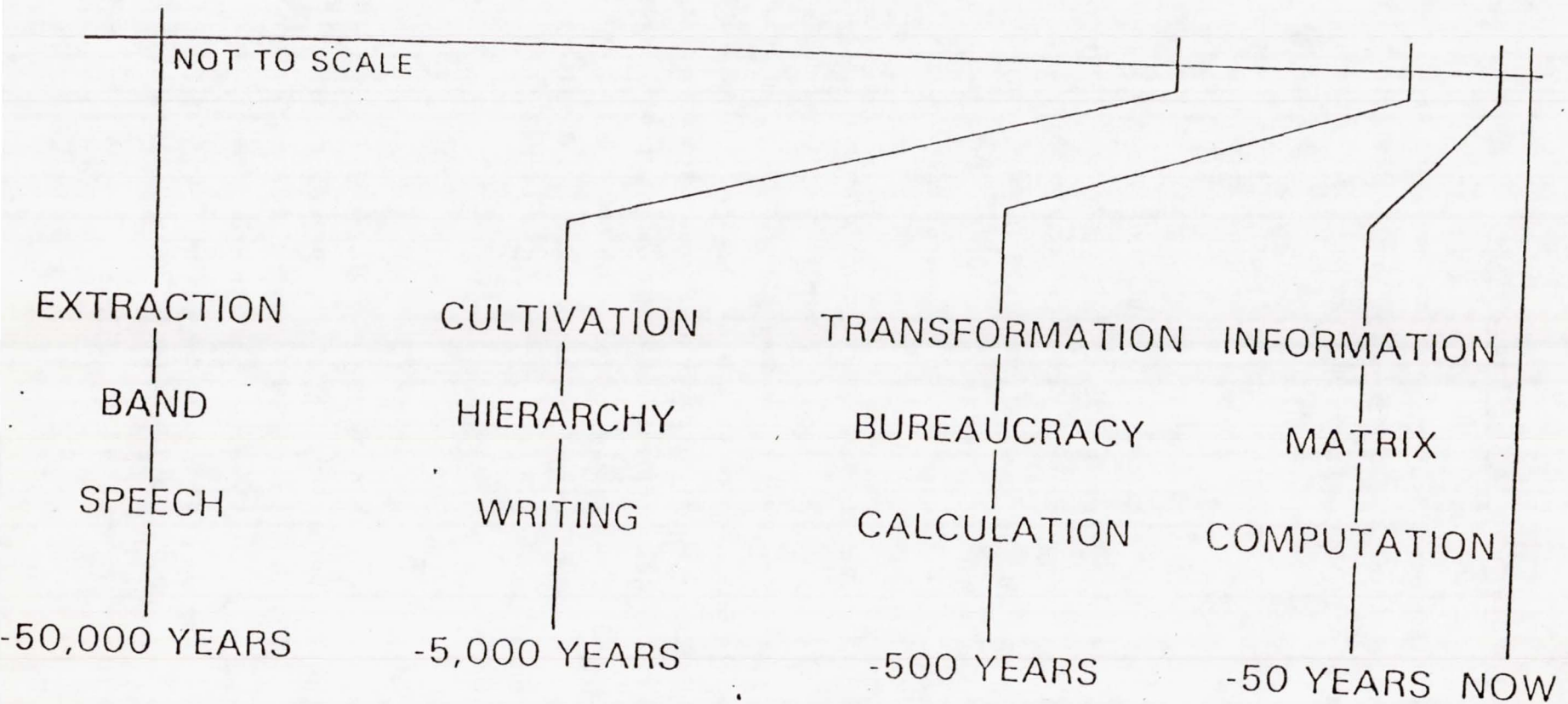


Figure 1

they hunted the available wildlife and gathered edible plants. They lived in small bands (30 to 50 people) with no firm organizational structure and no fixed and undisputed leaders. The communicative invention which set them apart from apes was, of course, speech. We should note, however, that speech -- language -- is not merely a way of communicating; it is also an instrument of thought, of mental calculation and reasoning (Vygotsky, 1962; Luria, 1976).

At the next level we find the ancient high civilizations -- Egypt, China, the Aztecs, and so forth. Cultivation has now become the principal economic mode of humankind; both plants and animals have been domesticated and are deliberately bred and raised for human purposes. These people no longer rely on nature to provide them with food -- indeed, they cannot, for hunting and gathering no longer can support the population densities of these civilizations (Harris, 1977). Hereditary hierarchies are the mode of social organization; to be a lord one must be the son of a lord; to be a guildsman one must be the son of a guildsman. Writing is a new means of communication at this stage, and leads to a substantial transformation in modes of thought (Luria, 1976; Benzon, 1978). Abstract thought becomes easier and a sense of history emerges.

The transition to "ancient civilization" took place approximately 5000 years ago, a phase of cultural evolution which lasted until approximately 500 years ago. At that time, the rich interchange of European, North African, and Asian cultures which had developed during the late Middle Ages catalyzed the European Renaissance, yielding what we know as "modern civilization." Here the reigning economic mode is transformation, the massive transformation of wood, coal, and oil into energy, the mechanized transformation of wide varieties of raw materials into manufactured goods. Transformation is accompanied by the bureaucratic hierarchy, which is explicitly divorced from the traditional hereditary entrance requirements (Ouchi, 1981). Instead, one earns a place in the hierarchy according to one's achieved competence to do the job.

The invention of the printing press is the communication advance generally associated with the Renaissance. While this device was essential to democratizing the cultural advances of the previous era, it did not have the

driving cultural transforming force of the invention of routine methods of calculation. The important historical figure here is the Moslem mathematician al-Khwarismi (Bernal, 1971; Fuller, 1981), an algebraist who consolidated Asian advances in mathematics and wrote a systematic text on calculation procedures which was translated into Latin about 1200 A.D.. These procedures are called "algorithms," a word derived from al-Khwarismi's name. With the use of algorithms it became possible to calculate accurate astronomical tables, to use those tables for navigation, and finally to calculate the profit or loss from trading ventures. These procedures thus made possible both modern science and modern accounting.

From calculation we move to computation and the modern "Information Age." The basic abstract foundations of computation were laid by Alan Turing immediately before World War II and by John von Neumann immediately afterward (Singh, 1966). By 1940 the percentage of information workers in the U.S. labor force equaled the percentage of industrial workers (almost 40% each), and in 1980 employment in the information sector was almost double that in the industrial sector (Oettinger, 1980). We are thus in an era when the production and control of information is the major focus of economic activity. Correlative to this we have the rise of decentralized administrative mechanisms such as the matrix (Toffler, 1980). Such organizations no longer have a single organizational hierarchy. Instead, they consist of multiple hierarchies which coordinate a structure of work groups that shifts with the changing tactical demands and needs of the organization. The increasing use of computers in management and communication functions (e.g., decision support systems, electronic mail and telconferencing) is likely to facilitate and accelerate this development by providing the flexible information control needed to manage shifting organizational structures.

With this evolutionary sequence in mind it is possible to contemplate the next logical event in the series of major human cultural developments. The Information Age may well also be the computer age, but the computer age is, at best, only partially unfolded. We know how to design hardware, but software is still a major problem (Rogers, 1980). The advances of Turing and von Neumann gave us the functional requirements a device must meet if it is to be a computer, but this has been of little help in the task of making software

production routine and reliable. It is quite possible that this job will require basic advances which stand in relation to computing as computing is to calculation, calculation to writing, and writing to speech. Such levels of advancement are the objects of intense, but as yet unsuccessful, pursuit by the artificial intelligence community.

Next consider the timing of these major advances in cultural evolution. While the numbers in Figure F-1 may be accurate only to an order of magnitude, the series of major cultural events appears clearly to be converging -- 50,000 years ago, 5000 years ago, 500 years ago, 50 years ago. This suggests that we may be nearing the beginning of yet a fifth phase in cultural evolution, one which might well be catalyzed by fundamental advances in software production (Hays and Benzon, 1981). This speculation is reinforced by a recent article in which Cesare Marchetti (1980) examines the cycles of invention and innovation over the last 300 years and foresees the beginnings of a new cycle which will create perhaps 100 new industries before the turn of the century. Many of these will be linked to information management, whether in genetic structures or computers. The changes ahead may well outstrip the magnitude of changes past. And the computer will be at the center of it all.

F.2 Planning for Computing

Organizational structure regulates the transfer of information and knowledge among people. At the lower end of an organization, people and machines act upon physical materials to produce goods and services. At the middle and upper levels of an organization, people regulate, transform, and communicate information throughout the organization. That is also what computers do -- regulate, transform, and communicate information. When the information is non-numerical, even state-of-the-art computers vastly inferior to humans, although they still play the same kind of role in an organization that people do. From this it follows that a coherent plan for introducing and upgrading computing technology in an organization must incorporate a model of the organization. Of course, this is only one component among several needed to think coherently about planning for computing.

It is convenient to organize these components into a five-dimensional

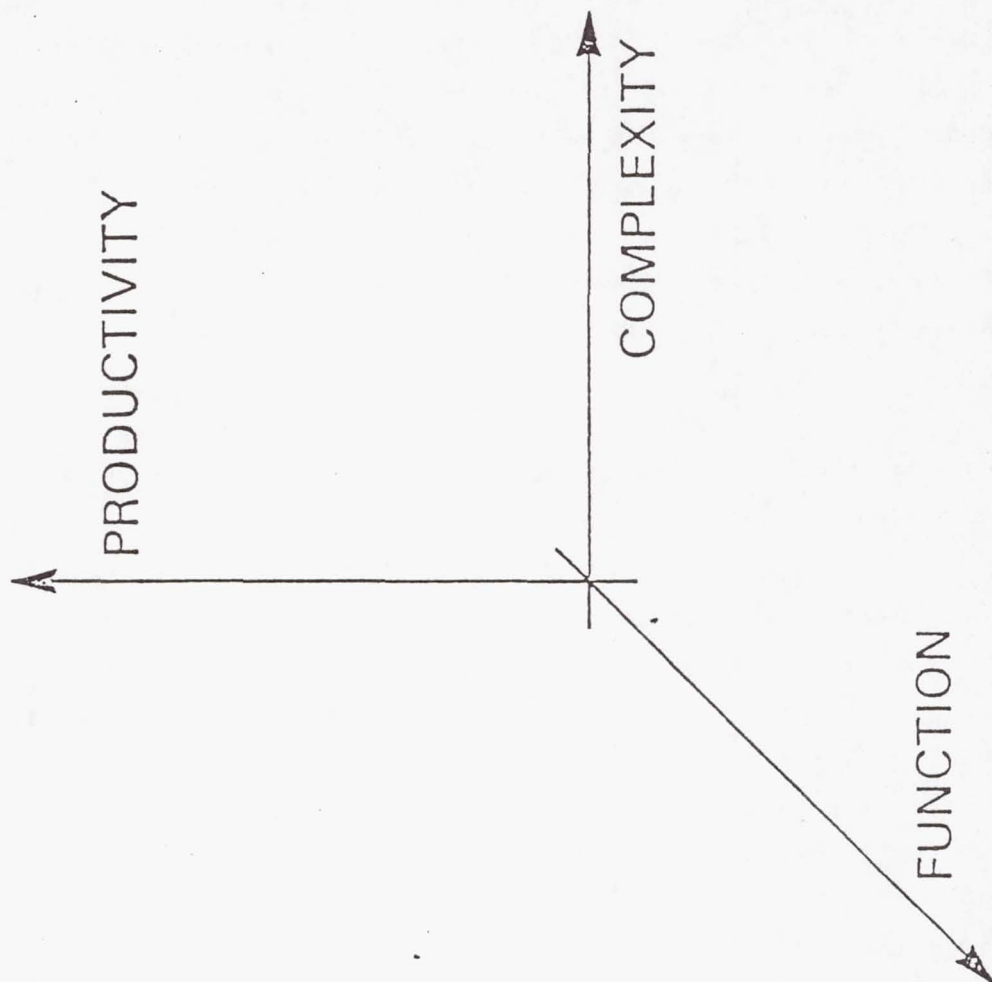
metaphorical space. The space is "metaphorical" because it is not here developed fully enough to be a proper formal model. In the present context it is intended only to permit the clear assessment of crucial relationships. One might well wish to use this metaphor in the preliminary design stage of a decision support system for computer planning, but that usage is beyond the scope of this Appendix.

Since it is difficult to imagine a five-dimensional space, it is convenient to unpack the metaphor into a three-dimensional component which can be embedded in a two-dimensional component. The three-dimensional component (Figure F-2) has axes for: (1) Productivity (of persons using a given set of computing tools); (2) Complexity of the computational domain, and (3) Functional type of computation. "Functional type" maps computing technologies into organizational structure. For a typical organization this might include marketing, manufacturing, financial, and research divisions -- in addition to overall management. The information requirements of these divisions are likely to be quite different and hence the technologies appropriate to each will differ as well.

The complexity axis represents complexity of the information processing required in a given function. "Complexity" ranges from routine functions, such as word processing, simple accounting, and large but logically simple calculations, through more clever functions such as decision support systems, image segmentation and classification, relational databases, to intelligent functions such as natural language question answering, machine translation, and automatic programming systems. Most of the computing technology in commercial use for the last 25 years has performed largely routine functions. The last few years have seen the introduction of clever computing into the commercial market and this will certainly increase during the present decade. Intelligent computing has seen much progress in the laboratory, but it is not yet sufficiently reliable and flexible to have any significant impact in the marketplace. Intelligent computing is, however, the leading edge of computer technology and is the area where new capabilities will be developed.

The final axis, productivity, represents the effectiveness of a given level of computational complexity in a particular functional role. How much

THE COMPUTING TRIAD



will word processing increase the productivity of a secretary? Will increased secretarial productivity be translated into a productivity increase for the bosses of secretaries? How much will a researcher's productivity be increased with a computer system for real-time monitoring of experiments? Will a decision support system result in increased management effectiveness? The general question being addressed here is how a given amount of money can be invested in computing technology to provide the greatest overall productivity increase. The answer to this question will require a fairly sophisticated model of the organization. The user will have to explore what types of computing technology can be bought for a certain amount of money, and the various ways each technology may be used within the organization. One organization might secure the greatest total benefit by investing in word processing equipment, another from a decision support system, while a third would do best with industrial robots.

It is not enough, however, to relate functional type and computational complexity to productivity. Organizations exist in time and they have goals. Computing technology is constantly changing. Planning for new goals and new methodologies casts the organization into the future. Hence, the three dimensions of function, complexity, and productivity must be embedded in a plane which encompasses both goals and time (Figure F-3). The temporal axis is needed to track, through time, the movement and mutual relations of both the organization and computing technology over time. Knowledge of the past can be used to project the future -- but if, as suggested in the previous section, mankind is about to enter a major new phase of cultural evolution, the future may not be sufficiently like the past to support very accurate projections.

The essential point is that computing technology is changing so rapidly that one must make that change a fundamental part of one's attitude toward the technology. In any given decision concerning the immediate acquisition of computing technology some consideration must always be given to the question of what will be replacing that technology in five, seven, or, at most, ten years' time. A given piece of equipment is likely to be outdated in five years and accurate projection of specific computing technologies is probably impossible beyond the ten-year frame. Where is the organization likely to be

~~PLANNING~~ ~~and~~ The Temporal Dimension
of Planning

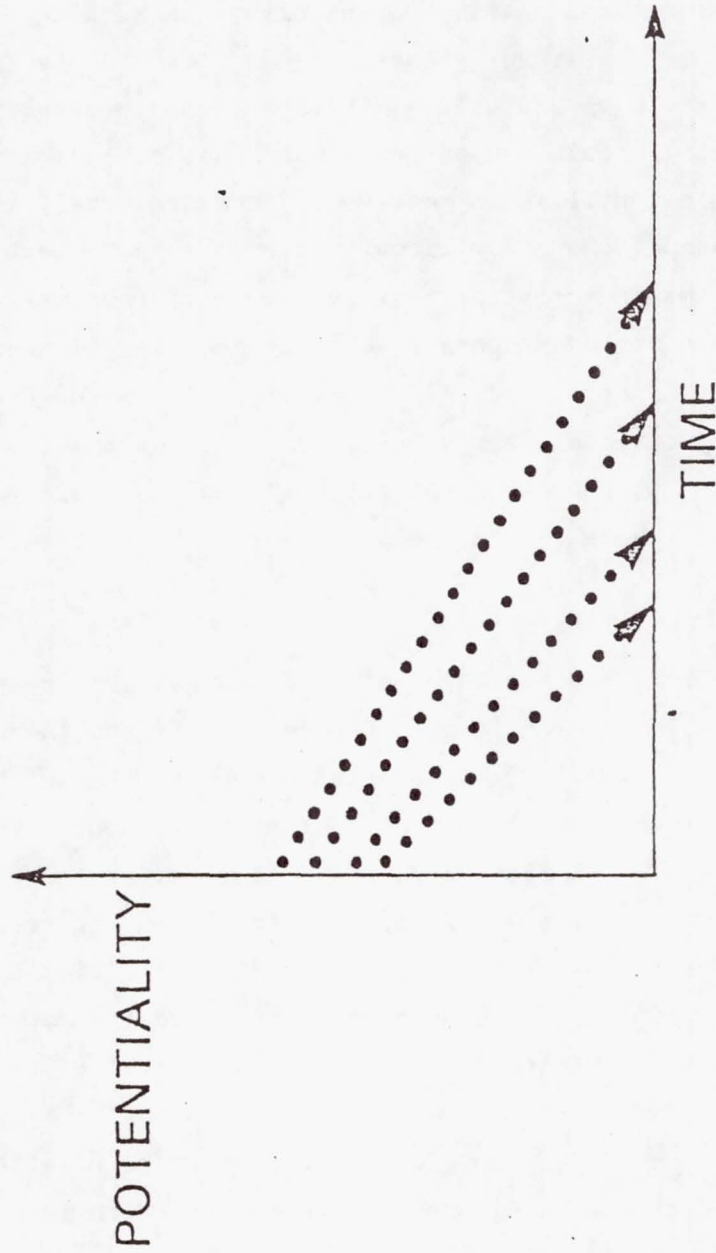


Figure 3

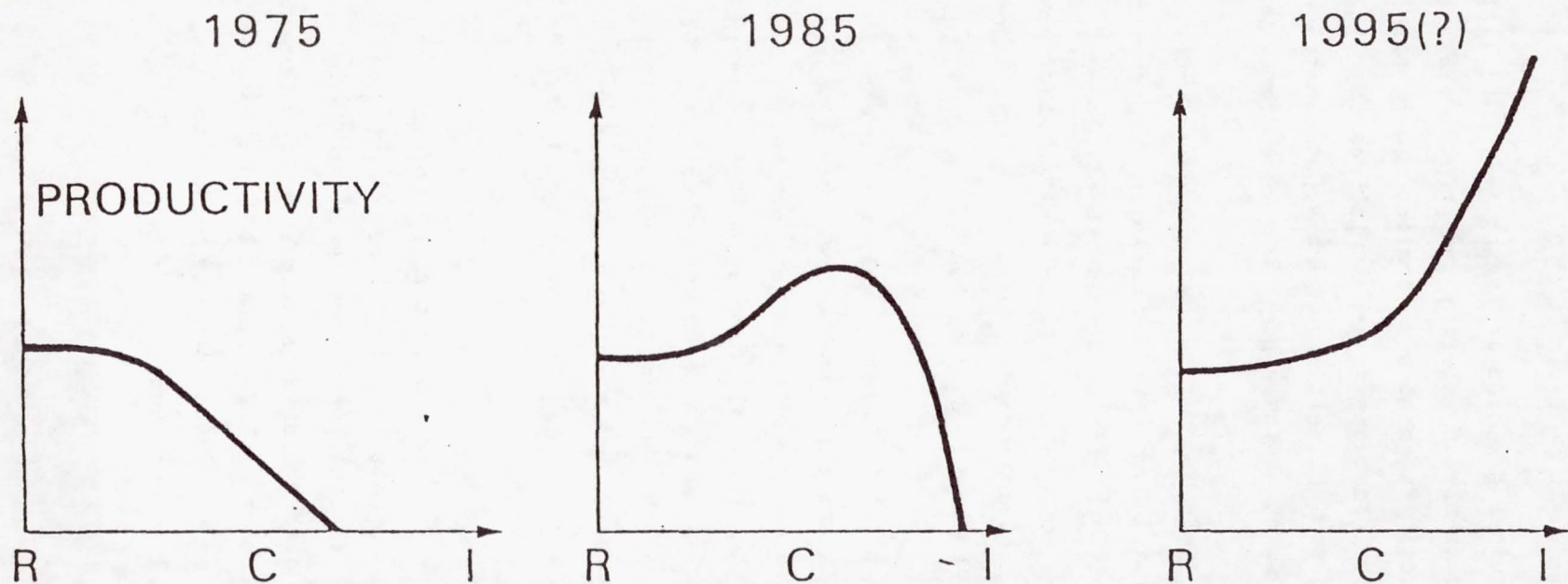
in five or ten years, what computing capacity will it need then, and which current acquisition will allow the most graceful transition to the projected need?

All such projections involve potentiality. Something with "O" potentiality is actual, in existence here and now. A piece of equipment to be phased out ceases to be "actual" -- it is assigned a negative value on the potentiality axis. Equipment on order which has not yet arrived remains a potential resource until it is received. Hardware merely identified as desirable is even further from actuality, even more "potential." Finally, it is possible to imagine computational systems which nobody yet knows how to design, much less build or operate -- these are even further from actuality. Hence the potentiality axis in Figure F-3 represents the difference between what is real and available and what is desired, imagined, or wished for. This dimension applies to general organizational goals as well as to computing technology (e.g., where should the organization be in five, ten, or twenty years?).

Any purposeful organization -- of cells, organs, or people -- formulates its goals at middle or high levels of potentiality and then operates over time to actualize that potential. At a given moment the organization is in a certain state. An essential part of that state is a plan incorporating the potential future of the organization. In fact, several potential futures are likely, both for the organization and for computing technology. The organization acts on one of them, and part of that action is to compare what is actually happening with what had been projected to happen. The difference between actual and projected can then be used to revise the plan.

Perhaps the most crucial projection concerns the relationship between productivity and the level of problem complexity which is manageable computationally. That is the relationship which holds the clue to the nature of the next phase in cultural evolution. The final section of this Appendix is a statement of potentiality which appears most likely today.

COMPUTING AND PRODUCTIVITY



PROBLEM COMPLEXITY

R: ROUTINE

C: CLEVER

I: INTELLIGENT

F.3 Computing and Productivity

Consider the three curves depicted in Figure F-4. The first represents the relationship between problem complexity and productivity as it was in 1975. Significant productivity gains were confined to routine computing -- numerical processing, simple databases, industrial robots. The programs performing these functions might well have been sophisticated and complicated for their time, but the functions they performed were simple and routine.

By 1985 clever technology may well be the dominant force, providing the largest productivity gains. Sophisticated decision support systems are beginning to have a significant impact already (Geoffrion and Powers, 1981), and program generators -- programming tools allowing relatively naive users to produce complicated applications programs -- will provide major assistance in the future production of software (Gordon, 1981).

The really significant shift, however, will not come until exceedingly complex problem domains fall within reach of reliable computing systems. These will be so sophisticated that they might just be, in some non-trivial sense, intelligent. Whether this will happen in 1995 -- as in the illustration -- or before, or after, is an open question. But unless there are heretofor undiscovered inherent limits to human intelligence, there is no reason to believe that someday such machines will not exist. When that day arrives, many of the computational wonders which are today the subject of science fiction may become reality.

Almost all computer programming might then be done by other computers. Most manufacturing will be almost totally automated and most manufactured goods will be custom-made for the purchaser. Much education will be achieved with the aid of sophisticated computing systems, making high-quality learning a continuous and pleasant part of life (Papert, 1980). Almost all mental and physical drudge work may be eliminated and the remaining jobs, with the aid of these computer tools, will become humane. And the people who work will gain tremendous productivity.

This future, of course, is hardly inevitable. Even if the technology

makes it possible, mankind may not desire it. What does seem inevitable is that someday someone will have to deal with these questions, not in potentiality, but in actuality.

F.4 References

Benzon, W.L., Cognitive Science and Literary Theory, unpublished Ph.D. dissertation, SUNY at Buffalo, 1978.

Bernal, J.D., Science in History, 1: The Emergence of Science, MIT Press, 1971.

Boulding, K., Ecodynamics, Sage Publications, 1978.

Fuller, R.B., Critical Path, St. Martin's Press, 1981.

Geoffrion, A.M. and Powers, R.F., "Management Support Systems," The Wharton Magazine, Vol. 5, No. 3, Spring 1981.

Gordon, R., "Applications Programmers: An Endangered Species?" Computer Careers News, Vol. 2, 6 April 1981, pp. 1, 7.

Harris, M., Cannibals and Kings, Random House, 1977.

Hays, David G. and Benzon, William L., "Metagram Software: A New Perspective on the Art of Computation, Final Report, RADC-TR-81-118, Rome Air Development Center, Griffiss AFB, New York, 1981.

Luria, A.R., Cognitive Development, Harvard University Press, 1976.

Marchetti, C., "Society as a Learning System: Discovery, Invention, and Innovation Cycles Revisited," Technological Forecasting and Social Change, Vol. 18, 1980, pp. 267-282.

Oettinger, A.G., "Information Resources: Knowledge and Power in the 21st Century," Science, Vol. 209, 1980, pp. 191-198.

Papert, K.C., Mindstorms, Basic Books, 1980.

Rogers, K.C., "Engineering Enters a New Cycle of Development and Definition," Science, Vol. 209, 1980, pp. 127-132.

Singh, J., Great Ideas in Information Theory, Language, and Cybernetics, Dover Publications, 1966.

Tatje, T. and Naroll, R., "Two Measures of Society Complexity," in R. Naroll, R. Cohen, eds., A Handbook of Method in Cultural Anthropology, Natural History Press, 1970, pp. 766-833.

Toffler, Alvin, The Third Wave, Morrow, New York, 1980.

Vygotsky, L.S., Thought and Language, MIT Press, 1962.

White, L.A., The Evolution of Culture, McGraw-Hill, 1959.

Appendix G. The Automated Work Environment: Opportunities for NASA

The office of the 1970's was not much different from that of the 1960's. Desks, chairs, typewriters, filing cabinets, telephones and an occasional duplicating machine were typical furnishings. Telephones now have more buttons allowing more options. Filing cabinets now include microfiche or microfilm capability. The typewriters are electrical and have some built-in memory. But as we enter the 1980's great and radical changes are taking place.

The typewriter is developing into a word/report processing system with extensions to electronic mail capability. The computer system which provides these functions can also maintain office files and provide access to decision support and management information systems. The telephone has gained teleconferencing capability, with the ultimate potential of extension to video capacity. Local area networks will connect offices at various locations. This Appendix will discuss these developments and what their impact is likely to be on the NASA office.

G.1 The Automated Office

There are four essential components of the office of the future: A handset/headset for verbal communication, a keyboard, a video-display unit, and a communications network for each of the previous units. The physical configuration of these units and how they are used differs according to the type of the office being considered, but they are the basic elements.

The handset/headset provides normal telephone communication. This might be a direct telephone conversation as we know it today. Or there is teleconferencing, which reduces the need for travel to conduct face-to-face meetings. A third use is to speak a message which is digitally recorded by the intended receiver's unit (if not answered by the person called), an extension of today's automatic answering systems. The handset/headset also allows verbal input to a computer which has the ability to understand spoken language. A digital recording of specific telephone conversations can be made and entered in to office database. Video capability could be added at a later date, though this is not necessary either for individual telephone

conversations or for teleconferencing to be successful.

The keyboard and video display units interact with a microcomputer which accesses a local area network connected to other offices, data processing facilities, storage devices, and gateways to other networks. The user, whether a manager, supervisor, engineer, analyst, secretary or clerk, can enter written material into the system, edit and review it, and then transmit and file it without a paper copy ever being generated. This use combines the formal aspects of the word/report processor with those of the electronic mail capability. A more informal use of "electronic mailboxes" supplements the use of telephone conversations, since the recipient need not be present to receive the message at his unit.

The keyboard and video display also connect users to the decision support, management information, or general computer system appropriate for his particular needs. Modern systems allow users to interact with a system either to produce what they desire or to permit "browsing" in order to consider various options before deciding. The display unit may have graphics capability, and visual material can be stored for inclusion with text. Hard copies can be generated if necessary.

The filing system may incorporate subject/keyboard structures but will be based on text processing algorithms for organization and retrieval. Not only can normal filing system requirements be met, but the system should also allow the preparation of historical summaries for deposition in the "organizational database," the organization's memory.

At the least, the units described need only the communication lines for connection to the appropriate networks. Users could work at home with only occasional visits to the formal office. This may permit the development of the electronic cottage industry concept popularized by Alvin Toffler and mentioned in Chapter 3 in connection with the future of NASA. Also, users traveling with their terminal in an attache case require only a communication line to be fully "plugged into" the office. Such an attache case, the handset/headset, display unit and keyboard are thus the "automated workstation" (see Figure G-1).

NASA TODAY



TOMORROW



PORTABLE WORK STATION

Figure G - 1

G.2 Using the Automated Work Environment

Although progress is being made, the systems described above are not presently available commercially. Further, once they are developed their introduction and use will be gradual. NASA must be prepared for the changes which these systems will bring, and prepared to integrate components into a coordinated system.

Today the individual Centers are progressing without coordination in developing word processing systems, electronic mail capability, and automated workstations. This Summer Study provided the first opportunity for many of the Centers and Headquarters to find out what the others are doing. It appears necessary for NASA Headquarters to provide some policy to ensure a coordinated growth of the automated workplace throughout the Agency. With a well-developed policy, NASA could be characterized as one of the more progressive government agencies in the use of computer technology.

Appendix H. NASA and Scientific Supercomputers

The need to solve various critical scientific and engineering problems during World War II led to the development of the "computer," which today has emerged as the single most important tool of science. As the capacities of computing systems have grown and unit costs have dropped, scientific users have attacked more and more ambitious projects. The computer can be used to model physical processes, control experiments, and automatically collect, analyze, and archive experimental data. For each of these uses there is a huge variation of scale, ranging from applications by individual researchers carried out on a desktop microcomputer to applications requiring teams of researchers executing immense codes on specially-built supercomputers. This Appendix focuses on these large-scale scientific computing requirements.

The commercial possibilities of the computer were recognized very early, and IBM with its general-purpose systems came to dominate the field. Most scientific problems can be readily performed on the general-purpose computer, but this is inadequate for scientific problems with extreme requirements. These extreme requirements may be a result of the need for extensive calculations in a modeling problem, for large data loads from the newest digital sensor technology (e.g., synthetic aperture radars, multispectral arrays), or for control of exceptionally complex systems such as a Shuttle launch.

NASA's scientific missions have given the computer considerably more importance within the Agency than could possibly have been anticipated in 1958 when NASA was created. Both aeronautical applications and space missions require extremely large-scale computations. Because of this the Agency has taken an active role in the development and operation of large-scale scientific computers, including the ILLIAC-IV (a pioneering array processor) and the five Shuttle computers with their voting schemes to assure reliability. Both projects involved innovative computer architectures, at the time of their conception.

There are several scientific and engineering problems within the Agency which demand computing capabilities of exceptional magnitude. Solution of

problems derived from the dynamics of large mechanical systems provides the bulk of large-scale scientific computation. In this context, a physical system may be viewed as a collection of huge numbers of "particles" with motions and characteristics too numerous to follow individually. Constructing a mathematical model of the system requires making one of two possible assumptions: Either the particles are considered to form a continuous fluid, or else a relatively small number of particles are tracked. Choice of either the fluid dynamic or the finite element method dictates which algorithms are used to implement the solution on a computer. The complexity of the resulting programs is also affected by the capacities of available computers, the amount of computational time allowed, and the required accuracy.

There are four principal advantages of using computer models in conjunction with actual experiments: Lowered costs, decreased turnaround time, reduced risk, and greater versatility. Nevertheless, use of computer simulations does not replace experimentation and theoretical analyses; rather, it interacts with both, leading to deeper insights by scientists.

A typical NASA application in fluid dynamics might be modeling the airflow around a body using the Navier-Stokes equations. This creates, in effect, a "mathematical wind tunnel." The finite element approach is used for Monte Carlo modeling of molecular motion and interaction in computational chemistry, and to describe individual segments of large structures to assist in structural design. Both methods influence computer design.

Specialized computers are also being developed to handle the unprecedented data streams that occur in some image processing applications. New generations of sensors -- synthetic aperture radars, multi-spectral arrays, and so on -- produce an extraordinarily high data rate which must be extensively manipulated to produce useful images and information.

The computer as developed in the late 1940's is characterized by its sequential operation. A single stream of instructions operates on a single datum (SISD), with, in most cases, program and data sharing the same memory. John von Neumann is credited with this early design -- indeed, the sequential architecture is named after him. This design approach has served as the

basis for almost all computers built. These machines are capable of doing all kinds of computing, provided that the processing can be done fast enough and cheaply enough. They are very well-suited to an era when hardware costs were high and reliability low, since they minimize complexity and hence the cost of the processing unit.

In the last 35 years the computer has changed from an unreliable, room-filling \$3 million behemoth to a \$30,000 desk-sized tool -- without sacrificing processing capability -- while at the same time, mainframe performance has risen exponentially. This revolution was aptly summarized by a recent ad in Computerworld magazine, which noted that if the automobile industry had achieved similar economies over the past 30 years a Rolls-Royce would today sell for \$2.50 and get 2,000,000 miles per gallon.

The main reason for the exponential improvement in performance is the development of faster and more reliable electronic devices. The relays of the 1940's led to the vacuum tubes of the 1950's, to transistors in the 1960's, and since then increasingly dense integrated circuits for both memory and logic circuits have been developed. Of course there are limits to the speed of silicon technology, and its successor has not yet emerged. The table in Figure H-1 relates critical parameters for typical commercial integrated circuit designs with the expected fundamental physical limitations.

Figure H-1. Commercial Integrated Circuits and Physical Limits

(Carver and Mead, 1980)

<u>Factor</u>	<u>Achieved by 1978</u>	<u>Anticipated Physical Limit</u>
Feature Size	6 microns	0.3 micron
Junction Delay	0.3 nanoseconds	0.02 nanoseconds
Switching Energy	10-12 joules	10-16 joules
Clock Period	30 nanoseconds	1-4 nanoseconds

The maturation of VLSI technology will reduce line width to about 1 micron and consequently allow faster operations, but the physical limits of silicon technology suggest that the next wave of improvement in computer capability will come from advances in computer architecture rather than electronics.

Architectural Options

Parallel operation is the most important current alternative to the von Neumann computer architecture. A computer doing ten operations simultaneously is, in a crude sense, operating ten times as fast as sequential machines using devices with the same switching speed.

The architectural revolution is already in progress. Starting with IBM's 360-9 and CDC's 6600 some parallelism has already been used in commercial SISD machines. Early implementations included development of separate I/O (input/output) and memory management processors, and some overlapping of instruction execution. However, there are a number of different approaches that involve parallel processing.

Pipelines

Execution of a single program instruction requires a series of machine operations. By increasing the complexity of the processor, internal steps can be separated and different parts of instruction lineages can run concurrently on successive data elements. This strategy is particularly effective in dealing with vector operands when the same series of operations must be performed on every entry in the vector. Both the Cray-I and Cyber 205 make use of several pipelines, with separate provisions for vector and scalar data.

SIMD

A single instruction stream can control several processing elements which perform the same operations on multiple data streams. This strategy provides an alternative approach to processing vectors, but it seems particularly well-suited for some image processing applications. The ILLIAC-IV, which is just now being retired from service at NASA-Ames Research Center, was a forerunner of a family of machines using this parallel approach. With any of the multi-data-path approaches, the manner of interconnecting individual processors and dividing the tasks among them becomes critical.

Multi-Computers

The multi-computer is a tightly linked network of independent computers each with its own memory. Each computer has its own part of the task to run, and communicates with other routines through the network. This is the most flexible approach since (depending on the program) the operation of the processing elements can be synchronized or allowed any level of independence. Presumably a multi-computer with 256 processors would be capable of somewhere near 256 times the processing speed of a single machine -- if it could be programmed efficiently. Problems that must be solved include the kind of network interconnection which will be most useful, and how to divide algorithms into independent segments for implementation.

There are other attractive features of this approach. It may be a way of exploiting the cheap, high-powered microcomputers developed using LSI technology, and some algorithms naturally lend themselves to being programmed in a highly modular manner. By implementing these types of multi-computers, some of the programming nightmare of task division can be avoided.

There is no doubt that machines can be built based on such novel architectures, and there is a flood of "paper computers" (proposed designs) emerging from university and industrial research groups. The use of (V)LSI technology can minimize the danger of unreliability due to increased complexity of the processor. However, there do exist significant difficulties.

The most serious hurdle to be overcome is that code developed for SISD machines will not maximize the potential of parallel architecture. Hence it is very likely that the algorithms themselves must be specially tailored to make best use of the architecture. The experience of users of current advanced scientific computers is that compilers do not efficiently use the machines. Existing programs are fine-tuned by hand, thus adding considerable time and expense to the development process.

To these difficulties must be added the problems involved with development of a new system. Bringing a new computer into production and commercially introducing it can involve as much as \$100 million. The evolution of a new architecture can take 10-15 years. The commercial market for large scientific processors is very limited and the easily foreseen software difficulties also

argue against development of new products. It is very easy for a manufacturer to decline to invest in such a high-risk enterprise. Both Burroughs and Texas Instruments developed large-scale scientific computers with novel architectures (BSP and ASC). In the first case, the machine was withdrawn without any deliveries; in the second case, several machines were delivered and then TI withdrew from the marketplace.

NAS

NAS is a computer designed to solve the Navier-Stokes equations using more than a million points in a three-dimensional grid. Use of this system of differential equations to describe fluid flows assumes that the particles of the physical system form a continuous medium. The algorithms to solve the system on a computer actually deal with different equations and involve many floating point calculations on long vectors.

Burroughs and CDC have performed design work for the array processor needed by the system. Burroughs' design calls for an array of processors, with all possible interconnections allowed through the use of an electronic cross bar switching network. The CDC approach is a multiple vector pipeline. Working with long vectors, either design would perform about a billion floating point operations per second. This is two orders of magnitude faster than the machines currently running simulation programs. NAS would be incorporated into a system with a large scientific mainframe (which would control its operation) and sufficient memory to store data about several variables at each of the grid points.

FEM

The assumption that a physical system consists of a finite number of particles naturally suggests a multi-computer architecture, with each small processor emulating a small segment of the overall structure. The Finite Element Machine (FEM), presently under construction for NASA-Langley, uses this approach to simulate aircraft structures. It will consist of a 6 X 6 array of TMS 9900 microprocessors each of which is connected to eight neighbors. Data on the forces acting on the segment will be routed along the

eight data paths to the processors standing in for the nearby pieces of the structure.

MPP

The Massively Parallel Processor (MPP) is an array processor based on the ILLIAC-IV, which could be included in an image processing system. The newest and most sensitive spaceborne sensors, such as the Thematic Mapper (TM) and Synthetic Aperture Radar, collect digital information which is transmitted through a digital communication network and then requires extensive digital processing to put it into usable form. Scenes from the Thematic Mapper, a satellite sensor system which collects data at seven frequencies in the visible and infrared spectrums will require about fifty operations per pixel to provide geometric correction. Total output of the TM can be as high as 10^{13} bits per day. The current technology allows processing only about 10 scenes per day -- TM can generate data for 400. Synthetic Aperture Radar will provide output requiring extensive manipulation to recover an image, on the order of 10^4 operations per pixel. Again, this is beyond current capabilities.

The MPP design is for a billion operation-per-second array processor tailored to the image processing problem. This is accomplished by a 128 x 128 array of one-bit processors operating at a 10 Mhz rate, controlled by a single instruction stream (SIMD), each with one thousand bits of memory. This is coupled with special staging buffers and extremely high-bandwidth buses to move data in and out of the array.

Z-MOB

Extensive unmanned exploration of space and the planets (with unforeseen hazards) and communication delays requires computers that can make local decisions in real time. Schemes which offer hope of leading to this kind of capability consist of a large number of co-routines that must interact extensively. The large amount of coordination makes execution of these programs on a SISD computer quite slow. But in a multi-computer, each routine occupies its own computer and data is transferred between routines by sending messages between processors, while processing is conducted concurrently.

Z-MOB uses this philosophy, connecting 256 Z-80 microcomputers to a ring network that allows data to pass between any pair of computers in one machine cycle. The programming difficulties of keeping the whole "mob" pulling together are formidable, but the possibility of getting a highly capable computer for under \$200,000 is very attractive.

H.3 Conclusions

Large-scale scientific computing is an area of great importance to NASA, but it is unlikely that the Agency's needs will be satisfied by an industry which is devoting most of its development work to the more lucrative commercial markets. This requires a continuing effort by NASA (in conjunction with the scientific community) to develop the kinds of processors needed and to make available the software tools needed to utilize them efficiently.

Appendix I. Computer-Aided Design, Manufacturing, and Testing

A CAD (Computer-Aided Design) system is an integrated hardware and software computer system established for the purpose of facilitating engineering design. Extensive use of interactive computer graphics is central to CAD systems, and high technology CAD systems represent the fastest growing market in computer graphics.

Design and engineering are at the heart of NASA's various activities, so CAD and graphics systems are being implemented within the Agency, albeit on a piecemeal, Center-by-Center basis. There are at least 33 CAD/graphics projects in the various Centers, ranging from the user of turnkey systems (purchased as a package from vendors) to specialized, custom systems, all the way to the development of integrated CAD systems.

This Appendix discusses the general capabilities and benefits of CAD; the terms CAD, CAM, CAT, and CAR; and the users of CAD in NASA. It also examines current and future areas of development including mechanical design, VLSI design, and database management and integrated design. Finally, a set of recommendations for NASA to act on now is given.

I.1 CAD Capabilities and Benefits

CAD systems range in capability from graphics systems to integrated systems incorporating basic graphics, drafting, analysis, testing, management of design databases, manufacturing, and operations (as represented in Figure I-1). The design database is the link between stages of the design process, between subsystems within the design, between users, and between manufacturing and operations. Existing CAD systems typically can reduce by a factor of four the amount of routine, labor-intensive work in drafting. Factors from 2-20 have been cited in the literature for various drafting tasks. For example, in NASA's LASS (Large Advanced Space Systems) program one person can perform a conceptual design and analysis in one day, so alternative conceptual designs can be generated in a timely manner (Garrett, 1981). Such productivity increases are essential in a time of shortage of engineers. More sophisticated existing systems could greatly increase capabilities, as for

COMPUTER AIDED SYSTEM ENGINEERING

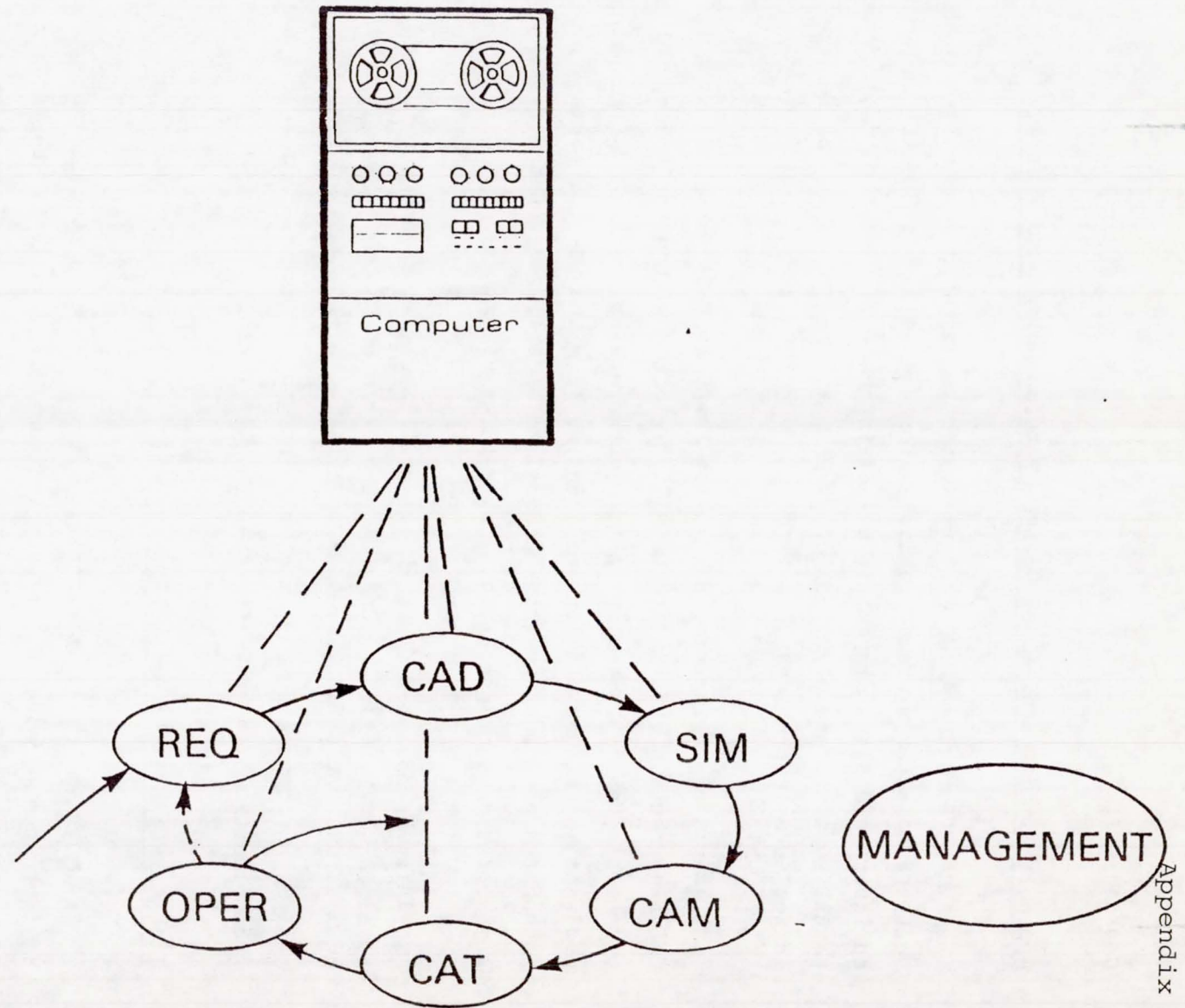


Figure I-1.

instance by creating three-dimensional drawings, rotations, and dynamic representations of motion.

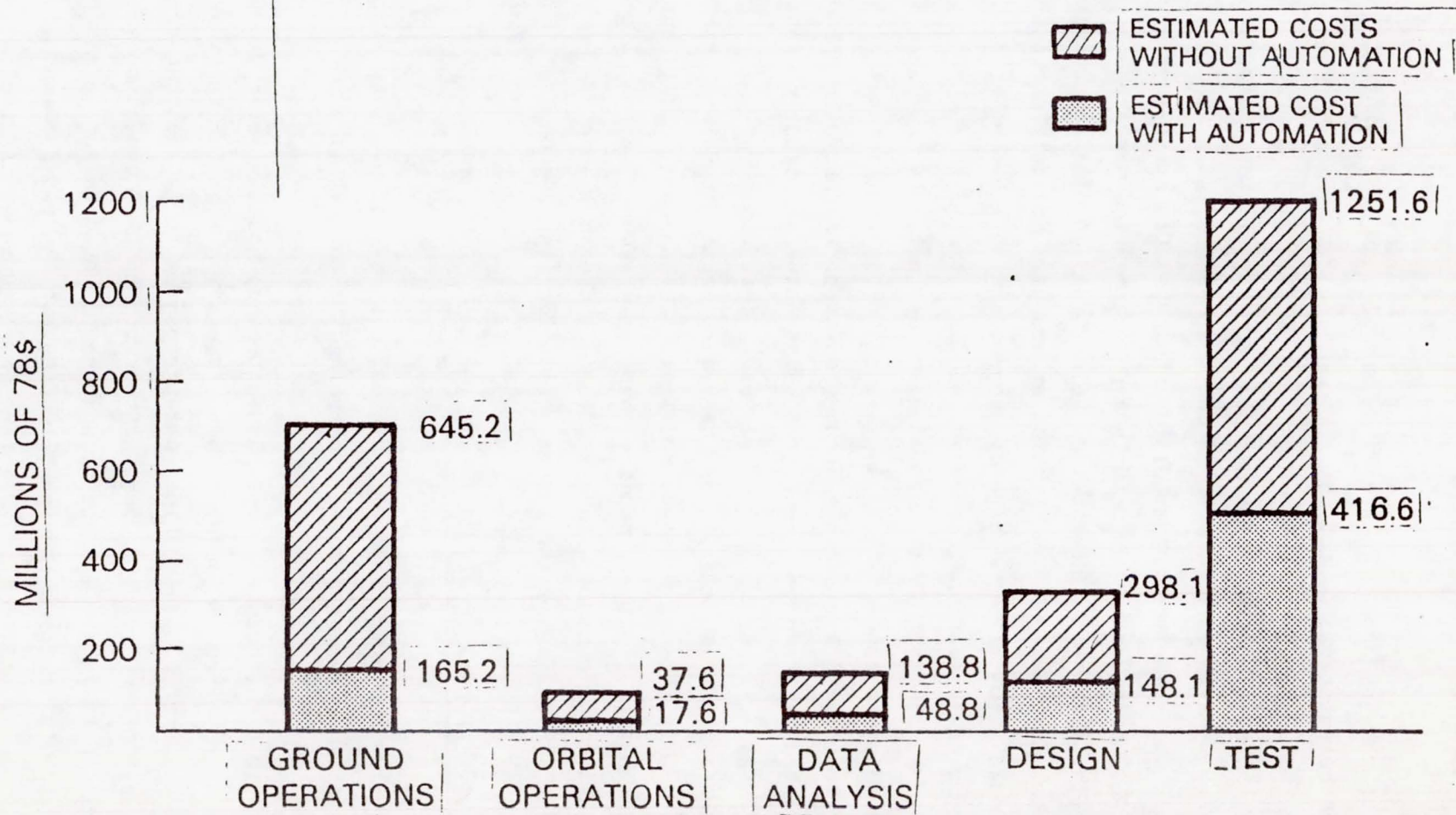
Special-purpose packages also greatly aid the interpretation of data from large-scale empirical and theoretical models, such as wind tunnel test results and molecular bonding models. Integrated CAD systems under development by NASA and industry would further reduce routine tasks and duplication of efforts (such as the time-consuming transfer of data between programs and systems), would allow checking by analysis and testing of the design, and would allow timely design of projects which would otherwise be too large and complex to coordinate, manage, and operate. Such projects include structures of unusual size which must operate very reliably in space, including autonomous spacecraft, space manufacturing facilities, and controllable, lightweight, large-scale structures to be assembled in space and then controlled from the ground. Such structures include microwave antennas, solar energy collectors, and telescopes. (See NASA's Forecast of Space Technology.) The only way to analyze and test a space structure on Earth is if it were in space is through accurate modeling. In the past, models have often been expensive and time-consuming physical mock-ups; in the future, computer-aided design and computer-based models can be used instead. Thus, substantial cost savings can be realized from automation (see Figure I-2).

I.2 CAM/CAT/CAR and Graphics

We use "CAD" here as the generic term for a variety of graphics-based system engineering tools, including the following:

- o **CAM** (Computer-Aided Manufacturing) usually refers to the use of computer programs in drawing up the specifications for NC (numerically-controlled) machine tool processing (e.g., setting up the NC instructions on a mylar tape to run the automatic machine tools). The design and manufacturing process could be set up so that the output of CAD becomes the input to CAM, but the two processes are not now always directly linked together. More advanced CAM includes the use of robots and other forms of automated manufacturing, and such CAD procedures need to be developed for NASA. For example, a teleoperated mechanical arm is

ESTIMATES OF YEARLY COSTS IN NASA OPERATIONAL ACTIVITIES WITH & WITHOUT INCREASED AUTOMATION (YEAR 2000, MILLIONS OF 78\$)



TOTAL WITHOUT AUTOMATION: 2471.6 | TOTAL WITH AUTOMATION: 896.6

SOURCE: A BENEFIT AND ROLE ASSESSMENT OF ADVANCED AUTOMATION FOR NASA

integral to manufacturing items in the Space Shuttle bay. Because of stress and bending of the structure, sensors must be devised to locate objects when relative positions can shift slightly. Advanced CAM systems also are central to space manufacturing and self-replicating systems (Long and Healy, 1980; Freitas, 1981).

- o **CAT** (Computer-Aided Testing) refers to the automated testing of engineering designs. The computer-aided testing may involve large-scale computational analysis, such as structural analysis using the NASTRAN finite element analysis program developed for NASA. CAT can also refer to the use of computer software to determine a test procedure and to run those procedures, such as testing of a logic design and circuit.
- o **CAR** (Computer-Aided Research) refers to the use of software tools to aid research, including storing and retrieving of data. For instance, results from empirical tests or theoretical models might be represented visually by the application of appropriate graphical analysis to the data.
- o **Graphics**, in addition to the above, may also be employed in command and control and used to visually represent the state of the system and to identify deviations from the norm or from some range of prescribed limits. Such graphics might be developed for satellite monitoring. Real-time image production involves the generation of visual displays quickly enough to monitor the current state of a system or to reproduce the system's behavior quickly enough to seem instantaneous from the user's viewpoint. Simulation for pilot training is one application. Non-real-time image generation may be used to realistically portray a mission, as in Blinn's color graphics film depicting the flight of the Voyager spacecraft. Image processing applications include near-Earth orbit satellite images, such as LANDSAT images.

I.3 Use of CAD and Graphics in NASA

The users of scientific graphics in NASA are primarily researchers and experimenters, so graphics is not as integral a part of their work as for a

mechanical engineer using CAD. For them, it is a data analysis tool. For both CAD tools and graphics, current users are primarily scientists and engineers who have learned to use these tools as they were found to be useful. However, for the systems to be fully utilized training must be available. This may involve formal classroom instruction or hands-on experience with very user-friendly systems having simple menus and instructions. Terminals must be readily accessible at all times.

CAD/CAM/CAT are being implemented on a piecemeal basis in NASA through the installation of minicomputers and turnkey systems (which are modified as necessary and as possible), and through the development of custom systems. Currently there is a lack of documentation for NASA's systems and little exchange of information within or among Centers on the software developed for these systems. There is some transportability of the software developed simply because of the widespread use of VAXs, which are well-suited to graphics applications. Of course, the problem of piecemeal development is not restricted to NASA but troubles American industry and agencies in general. More compatibility may result from the CORE (Computer-Oriented Reporting) guidelines of SIGGRAPH.

I.4 Hardware and Software Needs

Graphics has in the past been implemented by necessity on mainframe computers because these were the only machines with sufficient memory. The emphasis now is on free-standing minicomputer systems for graphics, possibly linked to a mainframe for large-scale computational analysis. There are advantages and disadvantages to minicomputers versus mainframes as well as for various CAD systems, graphics display devices, input devices, and output devices. Hardware costs are dropping and capabilities such as resolution and color for graphics terminals are increasing. The mainframe computer is computationally more powerful than a "mini," but as the system becomes saturated the response time may become too long. Minicomputers, although less powerful, also cost less, have less-stringent environmental restrictions during use, can be custom-tailored to CAD problems, and are well-suited to graphics applications. Size limitations can be at least partly compensated for by longer run times. For recent surveys of graphics and turnkey CAD

systems, see Bliss and Hyman (1981), Rosenberg and Fuchs (1981), and French (1981).

CAD systems and CAD-manufactured products that are developed by industry are aimed at large-volume markets. In the next few years, it is predicted that the CAD systems market will emphasize general applications and business applications, whereas NASA needs integrated systems and applications for special purposes such as Shuttle testing and large space structures. The industrial design and production of VLSI circuits, which requires CAD/CAM tools, is geared to large-volume production for economic reasons -- but NASA often requires prototype chips. The Agency cannot rely on industry to produce all of the CAD software tools and products it needs.

Turnkey systems, in contrast to "custom systems," are available from vendors as a ready-made free-standing configuration of hardware and software. Turnkey systems cost between \$20,000 and \$800,000, with a "typical" system in the range \$100,000 - \$300,000. The current CAD systems market is aimed at those who do not already have CAD systems, since such customers will generally experience the largest productivity gains from acquiring a new system. Some systems have been purchased by NASA Centers, but turnkey systems do not necessarily satisfy the needs of the Agency since many have only limited applications and no expansion capability. Accordingly, most CAD projects at the Centers involve custom systems for which NASA purchases the hardware and develops at least part of the software (see Appendix C).

The application of CAD is far more a software than a hardware problem, and the former is becoming increasingly important in developing high-level systems which integrate analysis and testing into CAD. The development of software can take several years, but due to the piecemeal development within Centers the software may not be transportable between Centers. Efforts should be made to generalize software tools and advertise them to potential Agency users. To encourage the development of more general tools, some means of directly supporting their development must be found. Currently, a manager has little incentive to spend, say, an extra 10% to make a tool his team has produced usable or available to NASA as a whole.

I.5 Using CAD: Areas for Development

Particular areas for development of CAD are in mechanical design, electrical design (particularly VLSI design), and database management and integrated design.

Mechanical Design

The focus of CAD in mechanical design is on the geometric representation of objects, usually as "wire frame" models. These are three-dimensional line drawings with segments corresponding to what would normally be hidden by "front" surfaces either being visible, modified in some way (e.g., dashed), or hidden. The treatment of hidden lines as other than visible involves a significant amount of storage and analysis as compared to having all the lines visible, and thus there is research in progress concerned with developing the most efficient means for storage and representation of such wire frames.

Work is also progressing in industry and universities on representations which can be analyzed as solid objects (i.e., having mass) so that, for instance, a sheet metal object is distinguishable in its properties from a solid object. With solid body models, mass and moments of inertia can be estimated and there can be automatic detection interference between components. A few solid-body programs already exist, but these require large mainframe computers and are considered still to be in the developmental stage (Krause, 1980).

There is also work proceeding on incorporating the analysis of the design into the CAD procedure, as for example doing structural, finite element analysis using NASTRAN (about 320,000 lines of FORTRAN code) or SPAR (about 30,000 lines) which were developed for NASA. There are also various proprietary programs available for CAD/CAM such as AD-2000 and its descendants. (AD-2000 Version 0.0 code has been purchased by IPAD with distribution rights to IPAD users.)

Computer-aided manufacturing has generally referred to NC metal-cutting or machine tool work. Processor languages such as APT have been developed. CAM,

however, is being extended beyond machine tool control to CAM for automated factories. The key step is the use of the computer to control more than one machine or more than one function (Gettleman, 1979). This will require the computer to process the appropriate sensory information and take the proper corrective action. Gettleman envisions a hierarchy of computers for such automated operations.

Electronics Design: Design Automation and VLSI

"Design automation" is the application of CAD to the design of complex digital systems. Currently if few chips are needed TTL (transistor-to-transistor) logic is used, for that involves low development cost at the expense of moderately high reproduction costs. In contrast, LSI and VLSI (large and very-large scale integration) involve high development cost but low reproduction costs. Thus, the focus of design automation for VLSI is to decrease the development cost.

Mathematically, many aspects of VLSI design can be represented as no-complete problems (Sahni and Bhatt, 1980; Donath, 1980). Given the complexity of the design problem, CAD tools are essential for logic design, circuit design, layout, and testing, but the development of advanced CAD tools will require advances in artificial intelligence. Some CAD tools, for checking design rules, testing logic rules, simulating the circuit, and other tests, are already available (Hon and Sequin, 1980), and the development of such tools is currently a major research effort in industry and academia. Still, the CAD/CAM tools actually used by a company to design hardware may be tailor-made to its architecture but then be cumbersome for the design process of new hardware. It has been estimated that it takes at least three years to design the CAD/CAT tools needed in hardware design.

With current approaches to VLSI design it is estimated to take 1.5-4 years from conception to final tested design, with the usual estimates being 1.5-2 years -- and the design process is costly. For these reasons, industry is geared to high-volume products to amortize development costs. Those users who do not require high volume are left to design and fabricate on their own or to hire out the design and fabrication (or just the fabrication) to a company

specializing in the production of custom chips. For design and fabrication to be readily separable functions, however, there needs to be an industry standard interface. No such standard now exists.

One design approach is to forego using all of the area of a chip that is optimally available and simplify the design process by dividing the problem into smaller pieces by using the PLA or master slice approach (Robinson, 1980). With another approach developed by Mead and Conway (1980), a student with no previous experience can perform a VLSI design within a four-month school term and have it produced as part of a multi-project chip (Conway et al., 1980).

It is important to involve not just the circuit designers but also the systems designers in VLSI design. It is interesting to note also that VLSI may make possible new computer architectures (see Appendix H) which will permit solution of larger design and analysis problems than are now feasible. For instance, to solve the no-complete design problem one can develop approximation techniques or "usually good" algorithms or use highly parallel algorithms with hardware designed to implement such algorithms.

CAD for VLSI is especially important to NASA because the Agency typically needs prototype, space-qualifiable, high-reliability chips. NASA needs special-purpose processors for signal, visible and radar image processing. Good CAD tools for VLSI design would allow the Agency to specify the necessary design to be manufactured by industry, possibly as part of a multi-project shipment or class run. (A class run occurs when one chip is produced to meet the special orders of several uses or users; the cost of the chip is then shared among the users, and each can take his copy of the chip and mask cut the unnecessary part.)

Database Management for Integrated Design

Database management is an integral part of any CAD process because of the large amount of memory required to store graphics information and design specifications. To implement integrated systems, the database structure and management becomes even more critical since the database is the link between

stages of the design process (e.g., conceptual, preliminary, and detailed design), between subsystems (e.g., structural, electrical, propulsion), between users (e.g., design engineers, project managers), and between design, manufacture, and operation of the system. A spacecraft has structural, electrical, and propulsive systems, among others, and it is necessary to ensure that the total system operates properly. One would like to do a variety of analyses on structure, propulsion, vibration, heat transfer, radiation, and buckling as well as checking cost, footprint analysis, mission timelines, and flight simulation and operations. Within a subsystem, there are a variety of levels. For example, a VLSI circuit can be variously represented by a register diagram, by a circuit diagram, by layout topology by mask geometry, and by a behavioral or test description. One would like to be able to access and change the appropriate level and also be assured that each level is consistent with all other levels. To conduct an analysis of the whole system there must be some means of transferring data from one analysis through pre- and post-processors or a database. A design database should in turn be linked into operations, where operations and control of an autonomous system would require the application of artificial intelligence to the construction of expert systems. But the necessary expertise and technology to implement such a database has not yet been developed.

A major development for CAD is the construction of a standardized database, and hence the development of associated specifications and standards. The enormous amount of data needed in the database will require new storage technologies (such as video disks) so that the data can be readily stored and still be quickly accessed.

NASA has taken the lead in the proof of concept of such a database in the IPAD (Integrated Programs for Aerospace-Vehicle Design) project. IPAD, a joint NASA-industry project, has as its purpose "to define and implement an integrated computer software system to support planning, data definition, and control of an integrated engineering design process, storage definition and control of databases containing large quantities of engineering data, and control and use of a large library of engineering application computer programs" (NASA IPAD Documentation, 1976). This project has been co-sponsored by NASA since it will help increase the productivity of an important

contractee industry.

There are three components of IPAD: Executive software, data management software, and geometry and utility graphics software. The components of IPAD are still being developed, but some programs are available. For instance RIM (Relational Information Management) is a working relational database, although it is not yet capable of the large-scale operations ultimately envisioned for IPIP (IPAD Information Processor). (See the IPAD references listed in Section I.7.) The IPAD project is coordinated with the Air Force Project ICAM (Integrated Computer-Aided Manufacturing).

I.6 Recommendations

For NASA to better utilize CAD in the near future and to develop capabilities for space utilization, NASA must: (1) Promote the exchange of information on the use and development of CAD; (2) strengthen its ties with researchers at universities; (3) develop CAD software; and (4) develop and promote methodologies for integrated design, such as Computer-Aided Systems Engineering (CASE). Each of these recommendations is discussed briefly below.

Exchange of Information

A major problem in NASA with respect to CAD is the lack of documentation for the systems as well as the meager exchange of information within and among Agency Centers on the software developed and the systems used. Hence there is duplication of effort and consequently less work on areas that have potentially great usefulness.

The acquisition of graphics hardware and software development should be coordinated by establishing a database/information system which can, for example, be searched to find software for particular applications. Such a system should be designed to encourage the interactive entry of new information into the system by the original developer or acquirer of the software. The system could perhaps be based on the teleconferencing and electronic mail systems that already exist outside NASA.

Within Centers, working groups should be established for informal discussion of the use and implementation of CAD/CAM/CAT/CAR and graphics. Examples include the CAD Steering Committee at JPL and CADRE at Langley. Workshops for NASA personnel pursuing related software projects at different Centers should be sponsored regularly. It is to be expected that the number of users and the use of graphics will remain widely distributed since the trend is toward minicomputers, application areas are distributed across the Centers, and long-distance graphics capability is currently impractical due to the high bandwidth requirements.

University Ties

Ties with universities should be strengthened to promote an exchange of information and state-of-the-art knowledge and to ease the Agency personnel shortage in CAD and computer sciences. Given the already large NASA investment in sophisticated state-of-the-art graphics and CAD/CAM systems (which are largely unavailable in academia), and the unique nature of some of NASA's design problems, it should be possible to attract visiting scholars and student interns to work on projects of mutual interest to NASA and the researchers. This is now being done on a limited basis by the large-scale computational chemists who use super-computers available only at Ames Research Center. NASA should also continue to actively participate in the newly formed CAD/CAM consortium of universities.

Software Development

NASA should directly apply and integrate existing CAD technology wherever possible, but the Agency should support development of those applications for which existing technology is insufficient and is unlikely to be developed independently outside NASA. The CAD and graphics technology areas to some degree all involve applications unique to NASA, and appropriate software will have to be developed for these (e.g., structural analyses of large structures in space, design and testing of space - qualifiable VLSI and LSI). Validation and verification procedures for the design -- and hence for the software underlying the design -- must be developed to ensure the accuracy of the analysis and the correctness and reliability of the design.

NASA does not need to "drive" the development of graphics hardware since there is a burgeoning industry already producing such hardware. In the near-term, the major innovations predicted for graphics are better hardware -- that is, color and better resolution for less money.

NASA should continue its support and for the development of computer architectures especially suited to the large-scale computations required for aerospace design. Examples include the NAS (Numerical Aerodynamic Simulator), the MPP (Massively Parallel Processor), and the FEM (Finite Element Machine) (see Appendix H).

Integrated Design: Computer-Aided Systems Engineering (CASE)

Further work must be done on database management in CAD and on integrating the analysis of the design into CAD. Significant effort is needed to tie the design process into the actual maintenance, repair, and operations of the craft designed. The design database can be tied into the operations database. Further, documentation and manuals should be maintained using a computer database to allow timely update of all appropriate instructions. Thus the design and operation of a system such as an autonomous spacecraft can be facilitated by a set of integrated computer-aided tools: CAD, computer-aided operations (e.g., expert systems), and database-managed documentation (Figure I-3).

I.7 References

Besant, C.B., Computer-Aided Design and Manufacturing, Halsted Press, NY, 1980.

Bliss, Frank W. and Hyman, George M., "Selecting and Implementing a Turnkey Graphics System." IEEE Computer Graphics and Applications, Vol. 1, April 1981, pp. 55-70.

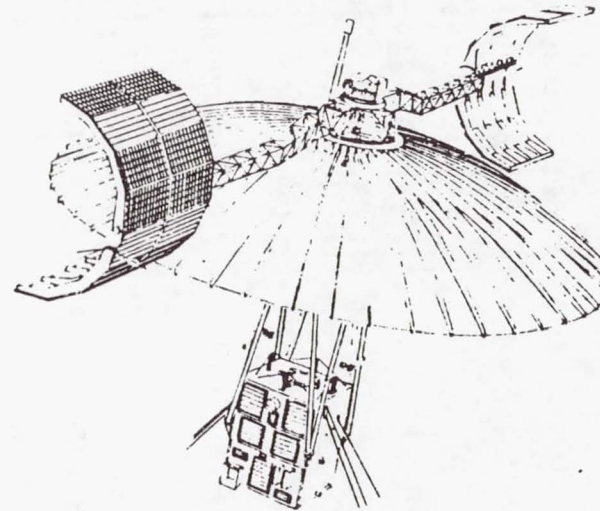
Burchi, R.S., "Interactive Graphics Today." IBM Systems Journal, Vol. 19, 1980, pp. 292-313.

CAD FOR SYSTEMS

DESIGN

BUILD

OPERATE



VLSI
DESIGN

MECHANICAL
DESIGN

SYSTEM DESIGN

GRAPHICAL DISPLAY
ANALYSIS & TESTING
OF DESIGN

CAM, e.g. NC TOOLING

TEST, GROUND, AND
FLIGHT OPERATION
PROCEDURES

INTEGRATED DESIGN → SPECIFICATIONS → DOCUMENTATION FOR
DATA BASE OPERATIONS

Conway, L., Bell, A., Newell, M.E., "MPC 79: A Large-Scale Demonstration of a New Way to Create Systems in Silicon," Lambda, The Magazine of VLSI Design, Second Quarter 1980.

Cottrill, H.E., "Computer Aided Design: A Feasibility Study," JPL 500-145, NASA-JPL, April 1979, and "Steering Group Interim Report," 11 November 1980.

Donath, W.E., "Complexity Theory and Design Automation," Proceedings of the 17th Design Automation Conference, Minneapolis, MN, 23-25 June 1980, pp. 412-419.

Erickson, Wayne J., "RIM -- A Relational Data Base Management System." CDC VIM 34 Minneapolis, 13-15 April 1981, pp. 1-80 to 1-85.

Fairbairn, Douglas G., "What to Expect from New VLSI CAD System." Lambda, Vol. 2, Second Quarter 1981, pp. 30-31.

Freitas, R.A., Jr., ed., Advanced Automation for Space Missions, Final Report of the 1980 NASA/ASEE Summer Study on the Feasibility of Using Machine Intelligence in Space Applications, University of Santa Clara, Santa Clara, CA, 15 May 1981, pp. 1200.

French, Hiram, "Trends in CAD/CAM Graphics Technology," Computer Design, Vol. 20, July 1981, pp. 99-102. ("Special Report on Computer Graphics")

Fulton, Robert E., "Overview of Integrated Programs for Aerospace-Vehicle Design (IPAD)," NASA TM 81874, September 1980.

Garrett, L.B., "Interactive Design and Analysis of Future Large Spacecraft Concepts," NASA Langley Research Center, Hampton, VA, AIAA Paper No. 81-1177, presented at the AIAA 16th Thermophysics Conference, Palo Alto, CA, 23-25 June 1981.

Gettleman, K.M., ed., "Fundamentals of NC/CAM," Modern Machine Shop 1979, NC/CAM Guidebook, Vol. 51, No. 8A, January 1979, pp. 250-406.

Hobbs, L.C., "Computer Graphics Display Hardware," IEEE Computer Graphics and Applications, Vol. 1, January 1981, pp. 25-32.

Hon, Robert W. and Sequin, Carlo, H., A Guide to LSI Implementation, Second Edition, Xerox Palo Alto Research Center, CA, January 1980.

Krause, John K., "Geometric Models for CAD/CAM," Machine Design, 24 July 1980, pp. 99-106.

Lewin, Douglas, Computer-Aided Design of Digital Systems, Crane Russak, NY, 1977.

Long, James E. and Healy, Timothy J., Advanced Automation for Space Missions, Technical Summary, 15 September 1980, NASA CR-163827.

Machover, Carl and Blauth, Robert E., eds., The CAD/CAM Handbook, Computervision Corporation, Bedford, MA, 1980.

McReynolds, Stephen R., A Benefit and Role Assessment of Advanced Automation for NASA, JPL, 14 April 1978.

Mead, Carver and Conway, Lynn, Introduction to VLSI Systems, Addison-Wesley Publishing Company, Reading, MA, 1980.

NASA IPAD Documentation: Executive Summary, D6-IPAD-70020-M, "Development of Integrated Programs for Aerospace-Vehicle Design (IPAD), prepared under Contract NAS1-12700 by Boeing, Seattle, WA; "Introduction to the IPAD User Manual," D6-IPAD-70023-M; IPAD Information Packet, submit requests to IPAD Program Support Manager, M.S. 73-03, Org G-5510, Boeing Computer Services Company, Box 24346, Seattle, WA, 98124; A Forecast of Space Technology 1980-2000, NASA SP-387, January 1976.

Robinson, Arthur L., "Are VLSI Microcircuits Too Hard to Design?" Science, Vol. 209, 11 July 1980, pp. 258-262.

Rosenberg, Jonathan B. and Fuchs, Henry, "Survey and Evaluation of Color-Display Terminals for VLSI," Lambda, Vol. 2, Second Quarter 1981, pp. 58-60.

Sahni, Sartaj and Bhatt, Atul, "The Complexity of Design and Automation Problems," Proceedings of the 17th Design Automation Conference, Minneapolis, MN, 23-25 June 1980, pp. 402-411.

Schneider, Jerry B., "Bibliography of Applications of Computer Graphics to Transportation Planning and Engineering Problems," IEEE Computer Graphics and Applications, Vol. 1, January 1981, pp. 155-158.

Storaasli, Olaf O., "Integrated Computer-Aided Design Using Minicomputers," paper presented at the ASCE Convention and Exposition, 27-31 October 1980.

Turn, R., "Computer Systems Technology Forecast," AD-A010 944 (avail. NTIS), The RAND Corporation, Santa Clara, CA, January 1975.

"Twelve Universities form CAD/CAM Consortium," IEEE Computer Graphics and Applications, Vol. 1, April 1981, pp. 114B-114D.

Appendix J. Expert Systems and Artificial Intelligence

Several studies recently have considered the potential utility of various artificial intelligence disciplines within NASA. McReynolds (1978) at JPL concludes that cost reductions could be achieved from a major commitment to extensive automation. McReynolds projects direct annual cost savings from reduced man-hour requirements of \$1.5 billion (in 1978 dollars) by the year 2000. (There are no explicit estimates of research and implementation costs required to attain the implied levels of automation necessary to effect such a savings, presumably because these are well below the expected return.) The report of the "Sagan Committee" (Sagan, 1980) focuses on machine intelligence in general and robotics in particular, and recommends that NASA quickly increase its capability in computer science. The 1980 NASA/ASEE Summer Study (Long and Healy 1980; Freitas, 1981) took a more futuristic look at the application of AI and robotics to NASA missions in space, and concluded that both were essential to NASA's long-range goals. Heer (1981) also has favorably reviewed autonomous systems in NASA's future from the JPL perspective. All of these studies concluded, explicitly or implicitly, that increased automation within NASA is both possible and cost-effective. In each case, specific Agency-related areas of application were examined and recommendations made as to how NASA might adopt particular technologies.

In the context of the present study and in view of the preceding work, it is both inappropriate and unnecessary to reassess the entire field of artificial intelligence and robotics. This Appendix focuses instead on a particular class of AI systems which has received insufficient attention in the past, "expert systems," an emerging software technology ready for application.

J.1 Relationship of Expert Systems Technology to AI

Much of AI is either only in the research stage -- clearly not ready for serious applications -- or is not proven technology, so that applications run a high risk of failure. However, expert systems technology, while entailing some risks, is sufficiently well-characterized to make it attractive in applications today.

During the 1950's, the notion of mechanization of the aspects of human behavior commonly called "intelligent" began to attract the interest of people in both psychological and computer sciences. By 1960, several interesting successes in accomplishing this had occurred. The psychology-oriented researchers were successful in modeling some aspects of behavior, and today this work continues and can be viewed either as a sub-field within AI, or, more commonly, as part of the discipline known as cognitive science. A related area of AI is concerned with the attainment of intelligent performance by computers, but is neutral with respect to whether or not the methods employed are similar to the workings of the human brain. This latter approach is often called the performance approach, and comprises the major effort in AI research which is of concern in the present discussion.

By 1960, optimism was high on achieving, in the very near future, mechanization of powerful and rather general-purpose intelligences which would prove useful in a variety of applications. Unfortunately, during the 1960's it became apparent that this goal would be much harder to achieve than originally believed, and by 1970 it was clear the near-term promise of AI would be limited to producing special-purpose systems rather than general-purpose intelligences.

The emphasis during the 1960's had been on the methods of inference that a mechanized intelligence would employ, called the power-based approach. Examples of power-based systems are the theorem-proving systems such as STRIPS (Fikes and Nilsson, 1971) and question-answering systems (Raphael and Green, 1968). Power-based AI systems gave the most promising successes in the 1950's and early 1960's but these successes were obtained by using extremely simplified domains. When more realistic, complex application areas were approached, it was found that "power-based" methods were in fact rather powerless. A new approach was needed, and by 1970 the first examples of "expert systems" began to appear. These are knowledge-based systems in which large amounts of high-level knowledge about the domain of interest is explicitly available to the system.

The knowledge-based approach is ready for applications in the real world. It has already produced several products which have demonstrated usefulness in

real-world situations, and the underlying reasons for their success are well enough understood to give confidence that the technology is indeed ready for use in producing new systems.

J.2 Introduction to Expert Systems

Expert systems are computer programs which contain a large amount of explicit knowledge about a restricted domain, and are able to apply this knowledge to achieve acceptable levels of problem-solving power in the restricted domain. The overall process of designing and implementing such systems, including the acquisition of the domain knowledge, is generally referred to as knowledge engineering.

In the past, AI researchers had attempted to produce competent problem-solvers by concentrating on the inferencing methods of the problem-solver, and paying relatively little attention to the form of the explicit knowledge. Such problem-solvers are described as power-based problem-solvers, since they depend on the power of the inferencing mechanism. Examples of systems taking the power-based approach are the theorem proving systems such as STRIPS (Fikes and Nilsson, 1971) and question answering systems (Raphael and Green, 1968). It has become clear that current and expected near-term technology for power-based problem-solvers is simply inadequate for use in most problem-solving situations involving real world complexity. During the last 15 years, the technology of the knowledge engineering approach has developed to the point where real world applications are possible.

The most common expert system architecture is based on the production rule formalism. This formalism consists of the following three parts: (a) A set of rules, (b) a database, and (c) a rule interpreter. Each rule is composed of an antecedent part and a consequent part. In a "pure" production rule system, the rule interpreter tries to find a rule whose antecedent part matches the database, in which case, the consequent part of the rule is used to replace, in the database, the item(s) matched by the antecedent of the rule.

As an example of a production system consider the following set of rules,

where the antecedent and consequent parts of the rules are sequences of characters:

1. 000 ----> bo
2. 010 ----> b
3. 011 ----> 01
4. 001 ----> b1
5. bo ----> b
6. b10 ----> R
7. b11 ----> F
8. b1 ----> A
9. 01 ----> R
10. 1 ----> R
11. b ----> R

Thus rule 6 says that if the database contains the consecutive characters "b", "1", and "0", then remove them and replace them by the character "F". For this example there is the additional restriction that the rules are allowed to match only the leftmost characters in the database. Thus rule 6 could match its antecedent to the database entry "b100110" but not to "1b100110." Also, the interpreter is required to select and apply only the earliest applicable rule in the list of rules.

Figure J-1 shows four example sequences of what the output databases look like for four different initial choices of database. A database containing "A" is interpreted as meaning "acceptance" of the initial database and a database containing "R" as meaning "rejection" of the initial database. The set of all strings of characters accepted by this production rule system are described as follows. A string is accepted if and only if it starts with "0", then has some number (possibly zero) or consecutive 1's followed by some number (again, possibly zero) or consecutive 0's, and then ending with a single additional "1". Note that without the restrictions of matching only the leftmost characters in the database and also using only the earliest applicable rule, this production system would not accept exactly the set of strings just indicated.

Figure J-1. Sample Process Sequences for Expert Systems

01 -----> A

rule 3 rule 3 rule 3 rule 9
01111 -----> 0111 -----> 011 -----> 01 -----> A

rule 3 rule 2 rule 8
01101 -----> 0101 -----> b1 -----> A

rule 2 rule 6
010101 -----> b101 -----> R1

Further variations might include:

- o The order in which rules are chosen may vary (this matters because more than one rule may be applicable).
- o The exact meaning of the notion of an antecedent of a rule "matching" the database can vary in several ways.
- o Instead of having the consequent parts of a rule simply be things to be added to the database, the consequent part may be a more general entity such as a piece of code to be executed.
- o The database may have various alternative structures, such as a set of elements or an ordered list of elements, and there may be probabilistic certainty information included for each element. Also, the database may have a hierarchical structure, and different rules may manipulate the database at different levels of the hierarchy.

The database contains all of the state-dependent information for the control of the rule applications -- that is, there are no program state control variables in the rule interpreter. This means that minor detailed features of the database are continually being checked in the process of choosing rules for execution. This gives production rule systems great sensitivity in dealing with domains of application that involve many special cases, and is one of the main reasons for the success of production rule-based expert systems in dealing with real world complexity in limited domains. (The benefits of this type of expert system are summarized in Figure J-2.) The alternative approach of procedural-based programming leads to such complicated software structures for the same real world complexity situations that it is in some circumstances impractical (Davis and King, 1977).

As attempts to automate systems become more ambitious, it is clear that hardware capabilities have progressed well beyond our ability to program complex tasks. Programs have been characterized as having two aspects -- logic and control (Kowolski, 1979). The "logic" part is the specification of what the program is to compute, but not the full details of how to compute it.




Figure J-2. Benefits of Expert System Technology

The overall benefits of using expert systems include those related to the use of computer science in general, namely:

- o Cost reduction
- o Increased capability
- o Quality improvement
- o Freeing personnel from mundane activities

Benefits more specific to production rule based expert systems are:

- o Increased portability of systems (by transfer of the knowledge base)
- o Systems more consonant with human thought processes than previous AI systems
- o Systems which are more understandable and can explain themselves
- o Systems with knowledge modularity and modifiability
- o Systems able to handle real world complexities at human level competency
- o Ability to handle domains for which there is no formal underlying theory

The control part is the additional information specifying how to compute what the logic part specifies.

The standard process in programming a complex task is to use a procedural programming language in which logic and control become inextricably enmeshed in executable code. This confusion of logic and control aspects leads to extreme difficulty in the implementation of complex tasks because:

- (1) The code has a tendency to be "brittle," in the sense that each piece can perform its task properly only when executed in a precisely defined context and when called by a precisely defined calling environment, any deviations causing erroneous computation; and
- (2) The logic of what is being done becomes lost in the control details of how to do it.

The procedural programming approach consequently becomes focused on the problem of understanding in detail the configuration, state of the program in terms of state control variables and overall program state. This requires that the task or computation to be implemented be analyzed and understood at a level of detail well beyond that which is really needed for the task itself. Factor (1) above leads to programs which are difficult to modify, are unable to explain their behavior to a user, and, along with factor (2), are difficult to understand. All of this contributes to the difficulty of developing such software.

The production rule formalism avoids many of these difficulties for some tasks and seems complimentary to the procedural programming approach (although neither is totally effective in replacing the other) (Feigenbaum, 1980). Each production rule represents a high-level piece of knowledge about the domain of interest. This gives modularity at the most appropriate level -- the knowledge level -- and enhances modifiability and understandability. In addition, an expert system using production rules can explain its behavior to a user by tracing the sequence of rule it is using.

J.3 Survey of Existing Expert Systems

The following is a survey of some of the better-known expert systems (ES). A variety of applications and internal architectures are represented.

MACSYMA (Martin and Fateman, 1971) -- MACSYMA is a system developed at MIT for the symbolic manipulation of mathematical expressions, solving equations and sets of equations, and other mathematical operations. MACSYMA is accessible over the ARPANET and has received a considerable amount of exposure to a diverse technical audience. It has earned a reputation as an effective pragmatic tool for problems in its domain.

DENDRAL, META-DENDRAL (Feigenbaum et al., 1971) -- DENDRAL is an ES in experimental chemistry which infers the structure of molecules based primarily on mass spectrogram data. There is also the ability to accept additional experimental evidence, such as information based on nuclear magnetic resonance spectra. DENDRAL is a production rule system which generates a list of potential molecular structures, and then simulates or models the operation of a mass spectrograph on each molecule, generating a simulated mass spectra. These calculated spectra are compared to the actual mass spectrum to select the most likely structure. The performance level of DENDRAL is comparable to that of a human expert, and thus DENDRAL has become an accepted working tool in several university and industrial laboratory environments. The META-DENDRAL program is an attempt to infer the production rules used in DENDRAL from experimental evidence.

MYCIN (Shortliffe, 1976) -- MYCIN is a medical consultation program which offers diagnosis and therapy advice for blood infections and meningitis. MYCIN developed after MACSYMA and DENDRAL were operational is the prototypical example of a production rule-based ES. The MYCIN system is cleanly separable into a set of production rules and an "inference engine." There are roughly 500 rule in MYCIN. The production rule are of the form: "If X holds, then conclude Y," where X is a set of elementary statements, and Y is a

single elementary statement. X holds only if every elementary statement in X is true, i.e., X is a conjunction of elementary statements. The inference engine of MYCIN is of relatively simple design, using the well-known method of backward chaining, where if the current goal is to establish that Y is true, this goal is replaced by the set of subgoals which are to establish the truth of each elementary statement in X. The name "EMYCIN" (Van Melle, 1979) is used to refer to MYCIN with its rules removed. By developing new sets of rules for EMYCIN, expert systems can be produced for other domains.

PUFF, MEADMED, SACON (Van Melle, 1979) -- PUFF is EMYCIN plus approximately 60 rules for interpreting pulmonary function test data. PUFF is currently operating in a realistic medical environment and produces test interpretations which are then reviewed by a physician. HEADMED is another EMYCIN-based system, working in the domain of clinical psychopharmacology. Approximately 275 production rules are used. SACON (Bennet and Engelmores, 1979), also based on EMYCIN, was developed to assist structural engineers in using a large and complex structural analysis system called MARC. MARC has many optional ways of analyzing an engineering structure (such as an aircraft wing or a nuclear reactor pressure vessel) and thus has required a large investment in time to learn how to use it. SACON acts as an interface system to assist the engineer in using MARC more quickly and efficiently.

NUDGE (Goldstein and Roberts, 1979) -- Scheduling problems are commonly encountered and handled by standard techniques (and are usually considered to fall into the domain of operations research). In certain situations, however, the scheduling problems are initially formulated in an ill-structured, informal, or incomplete manner, and it would be desirable to have the capability to handle these types of scheduling requests also. The NUDGE system works in the domain of scheduling meetings between personnel, and accepts incomplete requests. NUDGE knows about such basic things as time and place as characteristics of meetings, and also knows about and has the ability

to reason with such information as the role or function a particular person plays in a given meeting. For example, if Jack asks NUDGE to schedule a meeting with Jill for Monday, and Jill's schedule is full on Monday with other meetings, NUDGE has the ability to free Jill from certain already scheduled meetings by finding another person who can play the same role in those meetings as Jill.

CRYBALIS (Englemore and Terry, 1979) -- CRYBALIS is an expert system which uses electron density maps and amino acid sequence information to infer the three-dimension structure of protein molecules. Both the database and the production rule set have a hierarchical structure. The database contains, at its lowest level, a listing of geometric positions of the atoms that have already been identified. Other levels of the database identify composite structures, for example, "side chains" or "alpha helix." The three levels of production rules are (a) low-level specialist rules for identifying configurations, (b) task rules for selecting groups of low-level rules to be used together, and (c) strategy rules which are the highest level of control for focusing attention on various parts of the interpretation process.

PROSPECTOR (Gaschnig, 1979) -- PROSPECTOR accepts geological and sampling data from a mineralogical site and attempts to predict the presence and location of significant ore deposits.

SOPHIE (Brown and Burton, 1975) -- SOPHIE is an expert system used to teach electronics students how to diagnose faults in electronics circuits. Thus SOPHIE must be competent in both electronics and in the area of teacher-student interaction. In particular, SOPHIE has to model the state of knowledge of the student, and judge if the student is "testing" the electronic circuit in an effective manner relative to his current state of knowledge. SOPHIE does not use the production rule approach, and because of this it is weak in explaining to the user why or how it is making various conclusions.

GUIDON (Clancey, 1979) -- Guidon is an expert system of computer-aided

instruction (CAI) whose domain is the interpretation of production rules. Thus, GUIDON can be assistance in implementing the knowledge bases (that is, the production rule sets) of other expert systems by giving the program a current copy of the production rules of the developing knowledge base as input.

AGE, EXPERT (Nii and Aiello, 1979; Weiss and Kulikowski, 1979) -- These are both expert systems whose domain is the task of generating (designing and implementing) expert systems.

TEIRESIAS (Davis, 1976) -- This ES performs interactive analysis between a human expert and MYCIN to aid in debugging existing MYCIN rules.

SU/X (Nii and Feigenbaum, 1977) -- SU/X interprets spectral lines of input to understand the motions of physical objects in space.

MOLGEN (Martin et al., 1977) -- MOLGEN is used to plan experiments in molecular biology, specifically experiments on DNA.

J.3.1 Applications Areas for Expert Systems

Figure J-3 lists some areas of application of the expert systems which were used as examples above. Each system is listed with a single primary application, but most have multiple application possibilities. For example, META-DENDRAL could also be included under "construction of expert systems" since it infers rules which can be used by DENDRAL. SOPHIE could be used to help diagnose electronic faults in a real environment, instead of using its diagnostic analysis capability to support a teaching application. Figure J-4 is a listing of areas of potential application in terms of categories at the same level of generality as Figure J-3. Virtually any specific existing or imaginary expert system is likely to fall into more than one applications category.

Figure J-5 represents a more specific set of applications for expert systems, emphasizing those areas most likely to be of particular interest to NASA. For instance, previous studies have identified the area of ground

Figure J-3. Application Areas of Some Existing Expert Systems

<u>Mode of Use</u>	<u>Examples</u>
Consultation-Diagnosis	MYCIN, PUFF, PROSPECTOR, HEADMED
Technical and Research Assistants	DENDRAL, MACSYMA, CRYSLIS, META-DENDRAL, MOLGEN
Management (of personnel)	NUDGE
CAI	SOPHIE
Perception	SU/X
Construction of Expert Systems	GUIDON, AGE, EXPERT, TEIRESIAS

Figure J-4. General Areas and Modes of Use of Expert Systems

- o Autonomous control and maintenance (spacecraft, ground operations)
- o Consultation-diagnosis
- o Interpretation of sensing data
- o Anomaly detection and highlighting, management by exception (image interpretation)
- o Technical and research assistants
- o Management of: personnel, projects, facilities, contracts
- o Interfacing - computer to human and human to environment
- o CAI and simulation training
- o Management of expert systems

Figure J-5. Potential Applications of Expert Systems In NASA

- o Ground support operations for earth orbit and planetary spacecraft, and shuttle space operations.
- o Shuttle scheduling and monitoring for shuttle turnaround (e.g., payload scheduling).
- o Fault analysis and management for shuttle and other space systems.
- o Monitoring and management of astronaut health.
- o Scheduling of meetings for personnel.
- o Software development aids, both programming tools and software project management tools.
- o Systems development aids (computer aided systems engineering).
- o Research aids and analysis aids, such as DENDRAL and MACSYMA; and experiment scheduling, design, and interpretation aids; sophisticated real-time control of experiments.
- o Image interpretation; anomaly detection for further analysis by humans.
- o Automatic management to technical documentation.
- o Training of flight personnel; e.g., flight simulators for pilots; spacecraft-failure management training for ground support personnel.
- o Air traffic controller aids.
- o Policy analysis for NASA management.
- o Self-managing databases.

operations automation as an area where cost reduction or capability increase, or both, are currently feasible. The procedures for tracking, command sequence generation, and data processing are labor-intensive at present, and extensive automation seems practical with current AI technology. Efforts at JPL to automate the process of uplinking commands to a spacecraft is a first step. In the future there is potential for efficiency increases through the use of expert systems in the process of generating and verifying command sequences, and in overall control of the management of uplink activities.

For the Shuttle, economic considerations require rapid orbiter turnaround. This requires careful control of a large-scale operation which has several potential bottlenecks. The use of a sophisticated automated scheduling, monitoring and control system may be mandatory to achieve the required degree of control for rapid turnaround. The current scheduling approach involves the use of a computer-based, interactive, but people-driven scheduling system -- the computer serves primarily as a communication device between many people who collectively build a set of tasks into the computer for each flight (Coe, 1977). The program does some consistency checking and verifies the authenticity of the information being entered, as well as performing some standard OR-type scheduling. However, this system has very little knowledge about the specific domain of the STS ground operations. As such, it is limited in its ability to control the overall process and is thereby reduced to being an electronic messaging system for people. As additional experience is gained with the ground operations scheduling task, it would seem appropriate to move much of the initiative in the scheduling process from individuals to the computer. This will require an automated system containing considerable information about the ground operations activities, and expert systems would seem appropriate for this task.

Fault analysis and management is another area particularly well-suited to the expert systems production rule approach. The nature of the fault detection process often involves scanning and interpreting data over a long period of nominal operation, not an appropriate activity for human beings. When a fault is detected, the process of isolating its nature naturally falls into a many-case analysis, which can be handled by production rules. Similarly, the control decisions required for managing the faulted system (in

the case of spacecraft which must remain in operation) is also a task that can be handled by expert systems.

Every area identified in Figure J-5 could be automated without the use of the expert system production rule approach. However, the complexity of the task of automating these areas by more conventional software architectures is very high. This results in limitation of the goals of such implementations, simply to keep the effort at a manageable level.

J.3.2 Current Technology Level of Expert Systems

The performance of several expert systems is impressive by human standards. DENDRAL is a competent interpreter of mass spectra data at the professional level. MACSYMA has also proven itself competent in a work environment. The medical consultation systems such as MYCIN and PUFF seem to be roughly at the level of performance of a human expert in the particular domain, but in this area the objective quality of the performance is difficult to assess. Instead, the comparison is to the human experts in a domain, independently of the issue of whether or not the human experts are effective in an objective sense. This is particularly true for PROSPECTOR (Gaschnig, 1979). The performance of SOPHIE is quite interesting in that two areas of competence, electronics and instructional, are successfully joined.

At present, the major stumbling block to further progress with rule-based expert systems is the large amount of work required to explicate the domain-specific knowledge. In many domains this knowledge exists only inside the heads of human experts, and is in a procedural rather than an explicit declarative form. The process of extracting this implicit information from experts and converting it into an explicit set of production rules is time-consuming, involving (in the past) extensive interaction between the human expert and the knowledge engineer. Efforts are under way to develop techniques to reduce both the amount of time and the level of knowledge engineering skill required to build expert systems.

Most of the expert systems that have been built are either interactive consultation systems or data interpretation systems. In such modes of use,

the human user can accept or reject the analysis of the expert system. Little work has been done in designing production rule expert systems for environments where close human interaction and review of results does not occur. These more autonomous modes of action are of interest to NASA, for example, in the case of on-board control of interplanetary spacecraft. In order to have confidence in rule-based expert systems for such applications, it will be necessary to develop techniques to formally verify their correctness.

There have been a variety of modifications for the basic production rule formalism and the expectation is that no single standard will evolve. The reader is referred to the literature for descriptions of these (Davis and King, 1977; Feigenbaum, 1980; Georgeff, 1979; and Rychener, 1979).

J.4 Opportunity for NASA

Expert systems are a new technology which now exist primarily in academic and private research institutions, but is likely to diffuse rapidly into the industrial aerospace environment. There are a number of areas of activity within NASA to which expert systems could be profitably applied. This is both a challenge and an opportunity for NASA. The challenge is for NASA to acquire sufficient in-house capability with respect to expert systems to be able to recognize and evaluate potential application areas of this technology, and to be able to adequately manage contracts where expert systems are involved. The opportunity for NASA is for the agency to become proficient in this technology now, and thus further enhance its reputation as a progressive agency with respect to innovations in computer science, and to use the experience with expert systems as a lead-in activity for a general increase in the Agency's capabilities in the field of artificial intelligence.

J.5 References

Aikins, J., "Prototype and Production Rules: An Approach to Knowledge Representation for Hypothesis Formation," IJCAI-6 (Proceedings of the 6th International Joint Conference on Artificial Intelligence), 1979, pp. 1-3.

Barstow, D.R., "Knowledge Engineering in Nuclear Physics," IJCAI-6, 1979, pp. 34-36.

Bennet, J.S. and Engelmores, R.S., "SACON: A Knowledge-Based Consultant for Structural Analysis," IJCAI-6, 1979, pp. 47-49.

Brooks, R. and Heiser, J., "Controlling Question Asking in a Medical Expert System," IJCAI-6, 1979, pp. 102-104.

Brown, J.S. and Burton, R.R., "Multiple Representations of Knowledge for Tutorial Reasoning," in D.G. Bobrow, A. Collins, eds., Representation and Understanding, Academic Press, NY, 1975, pp. 311-349.

Clancey, W.J., "Dialogue Management for Rule-Based Tutorials," IJCAI-6, 1979, pp. 155-161.

Coe, R., KSC STS Automated Ground Operations Scheduling Systems, Book 2.1, Vol. 91 of the GOMMS Data Processing Design Specification Documentation, K-STSM-10.3.91, Kennedy Space Center, NASA, 1977.

Davis, R., Applications of Meta Level Knowledge to the Construction, Maintenance, and Use of Large Knowledge Bases, Ph.D. Thesis, AI Memo 283, Computer Science Department, Stanford University, CA, 1976.

Davis, R. and King, J., "An Overview of Production Systems," in E.W. Elcock, D. Michie, eds., Machine Intelligence 8, Ellis Horwood (J. Wiley distr.), 1977, pp. 300-332.

Engelman, C., Berg, C.H., and Bischoff, M., "KNOBS: An Experimental Knowledge-Based Tactical Air Mission Planning System and a Rule-Based Aircraft Identification Simulation Facility," IJCAI-6, 1979, pp. 247-249.

Engelmores, R.S. and Terry, A., "Structure and Function of the CRYSLIS System," IJCAI-6, 1979, pp. 250-256.

Faught, W.S., "Modeling Intentional Behavior Generation," IJCAI-6, 1979, pp. 263-265.

Feigenbaum, E.A., Buchanan, B.G., and Lederberg, J., "On Generality and Problem Solving -- A Case Study Involving the DENDRAL Program," in Meltzer and Michie, eds., Machine Intelligence 6, Halsted Press, NY, 1971, pp. 165-190.

Feigenbaum, E., Knowledge Engineering: The Applied Side of Artificial Intelligence, Report HPP-80-21, Stanford Heuristic Programming Project, Stanford University, 1980.

Fikes, R.E. and Nilsson, N.J., "STRIPS: A New Approach to the Application of Theorem Proving in Problem Solving," Artificial Intelligence, Vol. 2, 1971, pp. 189-208.

Freitas, R.A., Jr., ed., Advanced Automation for Space Missions, Final Report of the 1980 NASA/ASEE Summer Study on the Feasibility of Using Machine Intelligence in Space Applications, University of Santa Clara, Santa Clara, CA, 15 May 1981, 1200 pp.

Friedland, P., "Knowledge-Based Experiment Design in Molecular Genetics," IJCAI-6, 1979, pp. 285-287.

Gaschnig, J., "Preliminary Performance Analysis of the PROSPECTOR Consultant System for Mineral Exploration," IJCAI-6, 1979, pp. 308-310.

Georgeff, M.P., "A Framework for Control in Production Systems," IJCAI-6, 1979, pp. 328-334.

Goldstein, J.P. and Roberts, B., "Using Frames in Scheduling," in P.H. Winston, R.H. Brown, eds., Artificial Intelligence: An MIT Perspective, Vol 1, 1979, pp. 255-284.

Heer, E., Report of the JPL Advocacy Group for Autonomous Systems Technology Research and Development, JPL 715-128, June 1981.

Konolige, K., "An Inference Net Compiler for the PROSPECTOR Rule-Based Consultation System," IJCAI-6, 1979.

Kowalski, R., "Algorithm = Logic + Control," CACM, Vol. 22, 1979, pp. 424-436.

Long, James E. and Healy, Timothy J., Advanced Automation for Space Missions, Technical Summary, 15 September 1980, NASA CR-163827.

Martin, N., Friedland, P., King, J., and Stefik, M., "Knowledge Base Management for Experiment Planning," IJCAI-5, MIT, Cambridge, MA, 1977, pp. 882-887.

Martin, W.A. and Fateman, R.J., "The MACSYMA System," in S.k. Petrick, ed., Proc. ACM 2nd Symposium on Symbolic and Algebraic Manipulation, Los Angeles, CA, 1971.

McReynolds, S.R., A Benefit and Role Assessment of Advanced Automation for NASA, JPL 730-13, 1978.

Nii, H.P. and Aiello, N., "AGE (Attempt to Generalize): A Knowledge-Based Program for Building Knowledge-Based Programs," IJCAI-6, 1979, pp. 645-655.

Nii, H.P. and Feigenbaum, E.A., Rule-Based Understanding of Signals, Memo Hpp-77-7, Computer Science Department, Stanford University, CA, 1977.

Post, E., "Formal Reductions of the General Combinatorial Problem," AM. J. Math., Vol. 65, 1943, pp. 197-268.

Raphael, B. and Green, C., "The Use of Theorem Proving Techniques in Question Answering Systems," J. ACM, Vol. 169, 1968.

Rychener, M.D., "A Semantic Network of Production Rules in a System for Describing Computer Structures," IJCAI-6, 1979, pp. 738-743.

Sagan, C. Chmn., Machine Intelligence and Robotics: Final Report of the NASA Study Group, JPL 715-32, March 1980.

Shortliffe, E., Computer-Based Medical Consultation: MYCIN, Elsevier, NY, 1976.

Van Melle, W., "A Domain-Independent Production-Rule System for Consultation Programs," IJCAI-6, 1979, pp. 923-925.

Weiss, S. and Kulikowski, C.A., "EXPERT: A System for Developing Consultation Models," IJCAI-6, 1979, pp. 942-943.

Appendix K. Software Engineering and Technology

This Appendix considers the role of computers within NASA and examines the software environment within the United States as it pertains to the Agency. Briefly it has been found that:

- o Software has improved greatly in the last decade since the term "software crisis" first appeared (Dijkstra and Hoare, 1972).
- o Techniques exist today for producing good, reliable software, but they are not always used.
- o The demands on software, both in the amount and in the complexity, are growing faster than the ability to produce good software.
- o Software costs are rising and have become the dominant part of computer systems.

This Appendix starts with a description of the phases involved in any software project, then examines the software environment at NASA. Some of the problems associated with producing good software are discussed next, followed by a section on the state-of-the-art, including techniques and tools. Conclusions and recommendations are described at the end but are briefly summarized below. NASA should:

- (1) Expand its efforts to test and evaluate new software techniques.
- (2) Monitor itself, industry, and academia for new developments in software engineering.
- (3) Encourage new developments in software engineering.
- (4) Plan for software exchange.

K.1 Introduction to Software

"Software" is a generic term used to refer to a set of instructions to be performed by some computing device. This includes machine code as well as high-level language programs. Software may come in layers (for example, a user may input instructions into a statistical package which is written in some programming language and run on an interpreter -- three software layers).

"Software engineering" is a discipline concerned with the construction and maintenance of reliable software in some cost-effective manner. It uses mathematics for analyzing and certifying algorithms; engineering to estimate costs and define tradeoffs; and management to define requirements, assess risks, oversee personnel, and monitor progress. Unlike a purely academic discipline, software engineering treats the issues of applying principles, skills, and art to the economical development of programs which will run reliably and efficiently on real machines. One of the objectives of software engineering is to provide metrics and methods for measuring software (perhaps in terms of complexity, reliability, robustness, efficiency, usability, and generality).

An important unifying concept in software engineering is the "software life cycle," which describes the sequence of phases that comprise software development and evolution (Figure K-1). These phases typically include, but are not limited to:

Requirements -- the detailed specification of what is needed.

Design -- the plan of the individual computer programs needed to accomplish the task.

Programming -- writing (coding) of the programs.

Testing and Verification -- running programs to see if they work correctly.

Maintenance -- fixing bugs in a program.

Modification -- changing programs to accomplish some new task beyond that originally planned.

Since software specification is often imprecise and since the demands on software change with time, backtracking to earlier cycles often takes place.

SOFTWARE LIFE CYCLE (due to Dr. Barry Boehm)

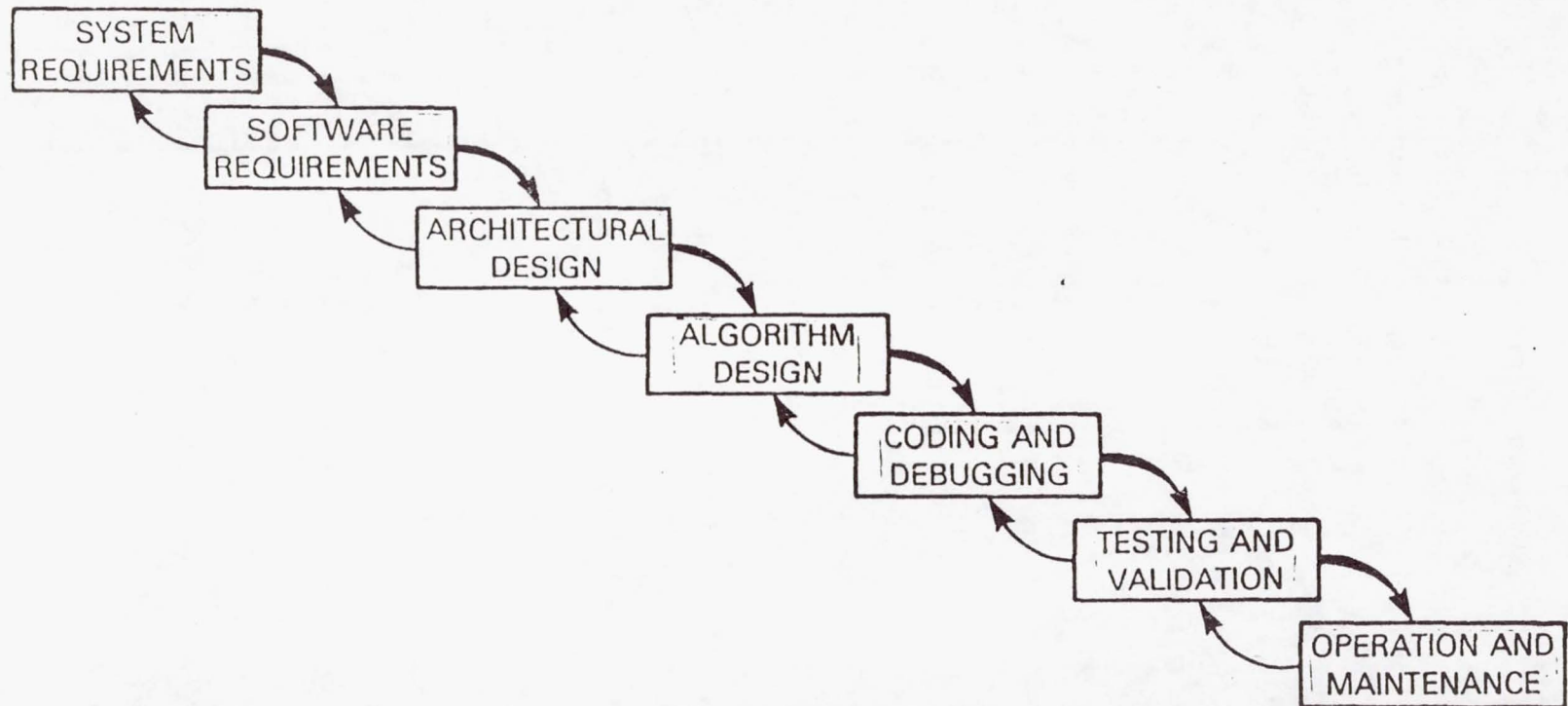


Figure K-1.

A SOFTWARE LIFE CYCLE IS USUALLY DEPICTED BY A "WATERFALL DIAGRAM" LIKE THIS. FAILURES IN ONE STAGE MAY FORCE A RETURN TO A PRECEDING ONE (COLORED ARROWS). ALTHOUGH THE CODING/DEBUGGING STEP (HIGHLIGHT) CAN BE THE FIRST ONE FOR SMALL PROGRAMS, IT OFTEN CONSUMES EXCESSIVE EFFORT BECAUSE PREVIOUS STEPS HAVE BEEN NEGLECTED.

It is not uncommon for several stages to co-exist and influence each other's progress. Thus it is important to be able to move both forward and backward from one phase of the life cycle to another and to review the progress (quality and quantity) of the work at a number of intermediate checkpoints. In this manner, it becomes possible to identify problems early in the project, take corrective actions, and most importantly to communicate the content and status of the development across groups.

It has been gradually recognized that creation of an integrated software development system involves the use of appropriate management and design skills, as well as organizational structures and the selection of a number of techniques that span the life cycle. Throughout the process, aspects of management and communication, including documentation, budgeting, personnel deployment, project review, scheduling, and configuration management, serve to tie the stages together and to provide the organizational environment in which the technical procedure can be made effective. While new technical procedures and modern management techniques can each improve the software production process, innovations acting on the combination of these areas can have a potentially greater effect on the software project as well.

K.2 Current NASA Status

NASA has a large inventory of software for many different computers, written in many languages. Software is developed for many uses, including (1) one-time or short-time use and (2) long-life usage to be passed on from mission to mission. Software may be developed directly by NASA personnel, by software contractors according to a specification, by contractors participating as a team with NASA personnel, or by contractor personnel as a byproduct of an engineering task. In addition, NASA acquires license to use proprietary software on its computers.

Many of NASA's software projects are large and complex. It is becoming increasingly important to understand the relationships between the parts of the system. Much of NASA's software has been developed as a subsystem to a much larger system. For example, the software for a flight project (on board spacecraft software as well as ground software). Much of this is custom

software for a particular system. Additional uses of the software for future projects requires modification to the existing software. In many cases these modifications can be substantial. If time and funds had existed during the original development, the software could have been developed with more flexibility to accomodate itself to many flight projects.

A major portion of the software in NASA is contracted to industry. Thus, NASA acts in large part as a manager of software projects. The relationship between NASA and a software contractor can be summarized using the categories in the following discussion.

Requirements -- Typically, NASA gives a specification to the contractor of what is to be done. For example, the Agency might want a program to compute midcourse corrections on a deep-space mission. Other things might be specified such as the programming language to be used, the reliability of the program, the speed and memory to be used. The contract might also include the way in which the software is to be created (e.g., top-down design, structured programming), the way in which progress is to be reported back to NASA (e.g., documentation, recording of milestones), and the way in which the program is to be tested once it is running. There are a few requirements languages available to assist in this process, but generally they are given in English.

Design -- NASA may or may not be involved in this phase. The concepts of a top-down approach and modular programs seem to be widespread and popular, but the process is still manual, and little help is currently available.

Coding -- NASA is usually not involved. There are several higher-order languages in use today and there are many aids available (compilers, debuggers, editors, preprocessors). Structured programming seems to be a popular and effective aid for programs written in a high-order language but not for programs written in machine or assembly languages.

Testing and Verification -- The contractor generally performs the initial testing. For crucial software, NASA often does its own testing and verification once the software is delivered. Testing is still largely a manual task and involves running the programs against test data to see if they

perform correctly. There are some aids available including path testing and static analysis (checking for type and whether variables have been referenced).

Maintenance and Modification -- These are done either by NASA or by the original contractor or sometimes even by a new contractor. These are difficult tasks because of the communication problems involved. Modifications often are not done by the original team and frequently there is time gap between the original project and the change. The phases involved include understanding the existing software, modifying the existing software, and testing and verifying the new software. There are few aids available for those tasks other than using the best people available.

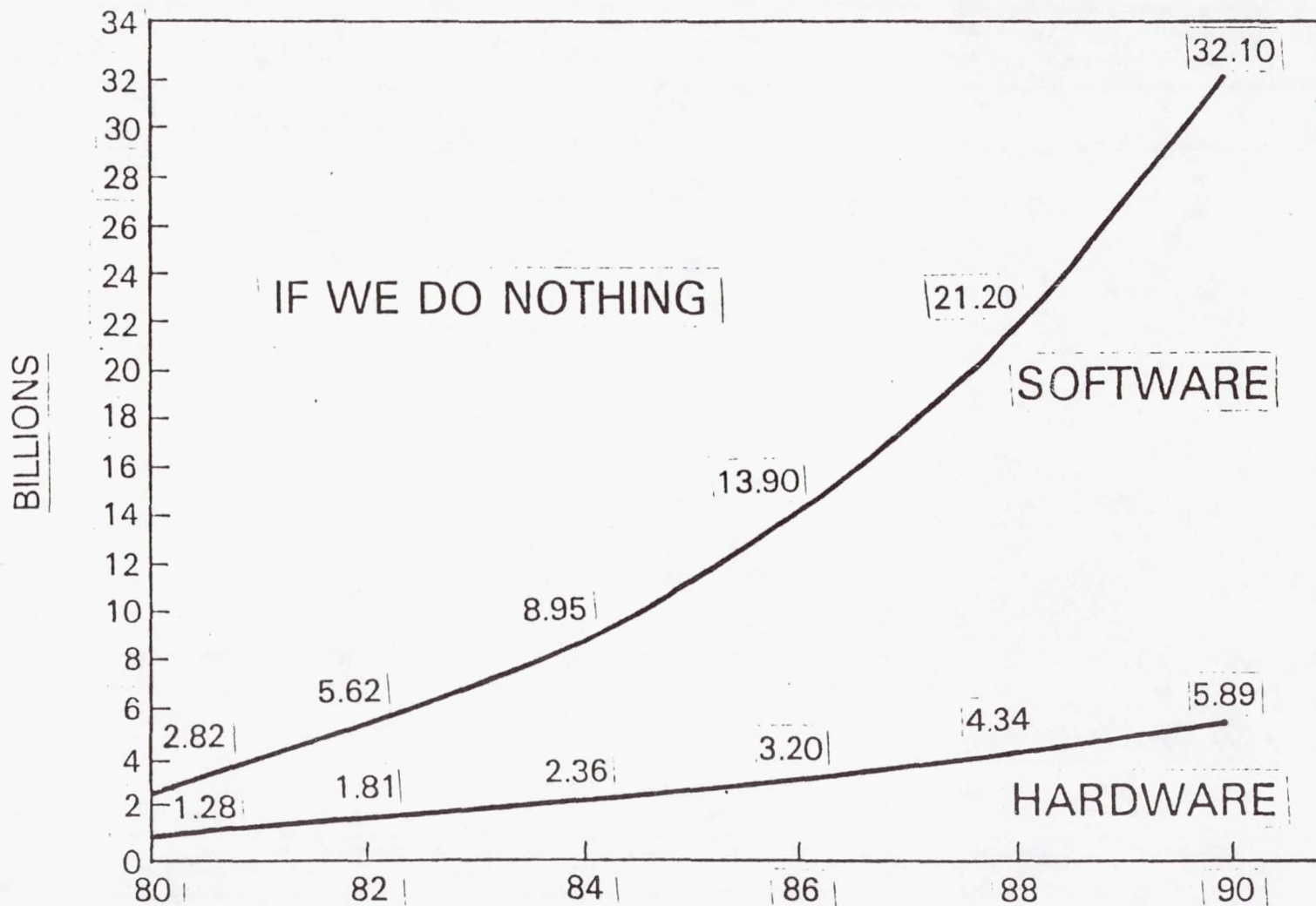
Software quality has improved drastically within the last decade. New management techniques and new tools have contributed greatly to this improvement, but still there is a wide range of quality in the software produced today. Currently one factor limiting improvement in software development at NASA is the inability of Agency personnel to use new techniques as they are developed.

K.3 Costs and the Software Gap

Software in the United States is a large and growing domain. While hardware costs have drastically declined, software costs have risen. A recent study of the software industry (Belady, 1979) projects a doubling every five years from 2% of the United States GNP in 1970 to over 20% of the GNP by 1995 if current trends continue (Figure K-2).

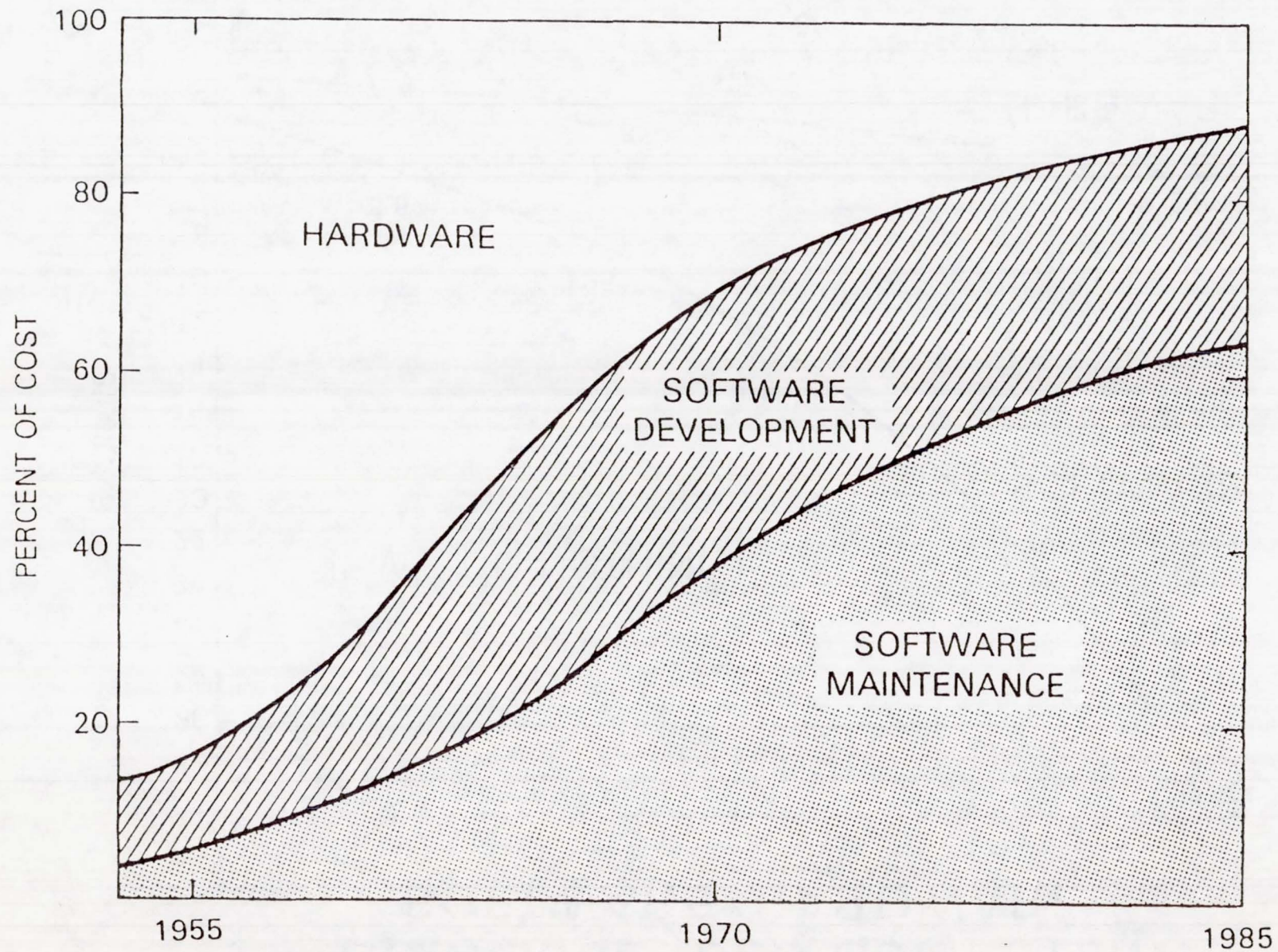
As hardware technologies improve at a spectacular rate, the problems of software development are changing. Computer systems will become widely available with greater processing power and more storage capability, with multiple microprocessors handling functions which were once software functions and with fault-tolerance in hardware systems providing more reliability. In general, these systems will provide more productivity and reliability for the hardware dollar. It is this hardware revolution that is putting the pressure on software to become more cost-effective and reliable.

DOD EMBEDDED COMPUTER SOFTWARE/HARDWARE

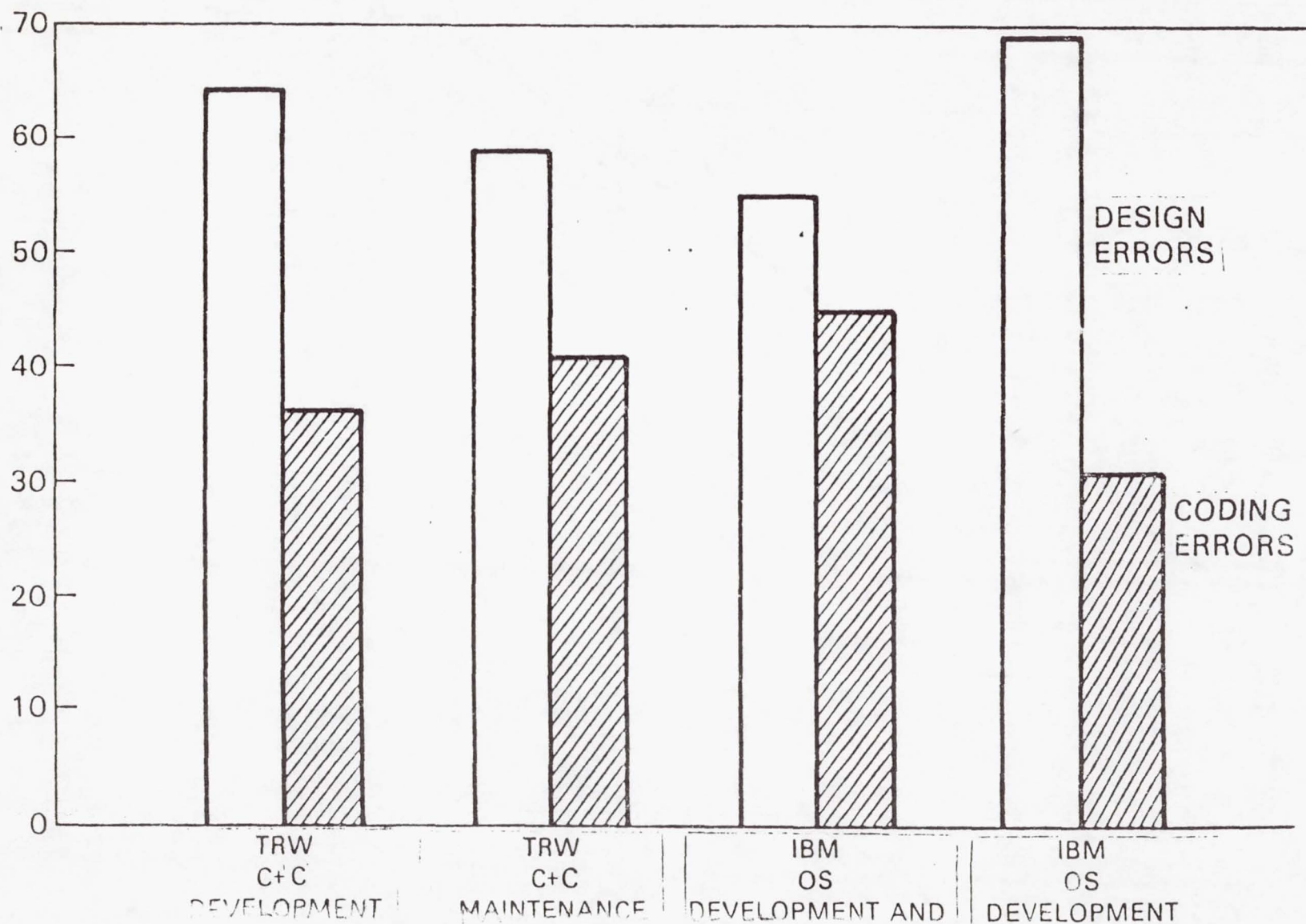


SOURCE: ELECTRONIC INDUSTRIES ASSOCIATION

HARDWARE-SOFTWARE COST TRENDS



MOST ERRORS IN LARGE SOFTWARE SYSTEMS ARE IN EARLY STAGES



THE PRICE OF PROCRASTINATION

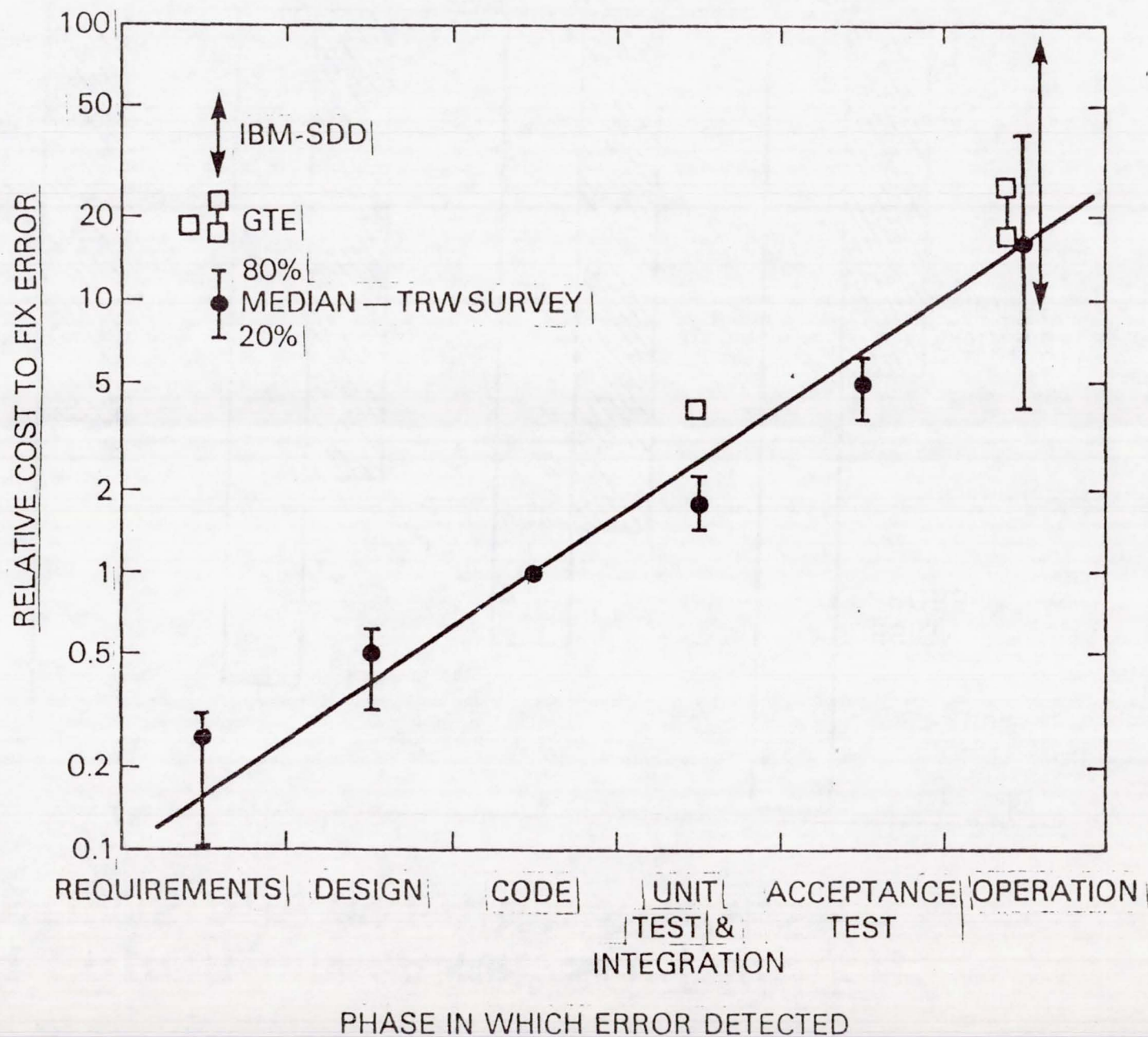


Figure K-5.

SOFTWARE LIFE-CYCLE COST BREAKDOWN

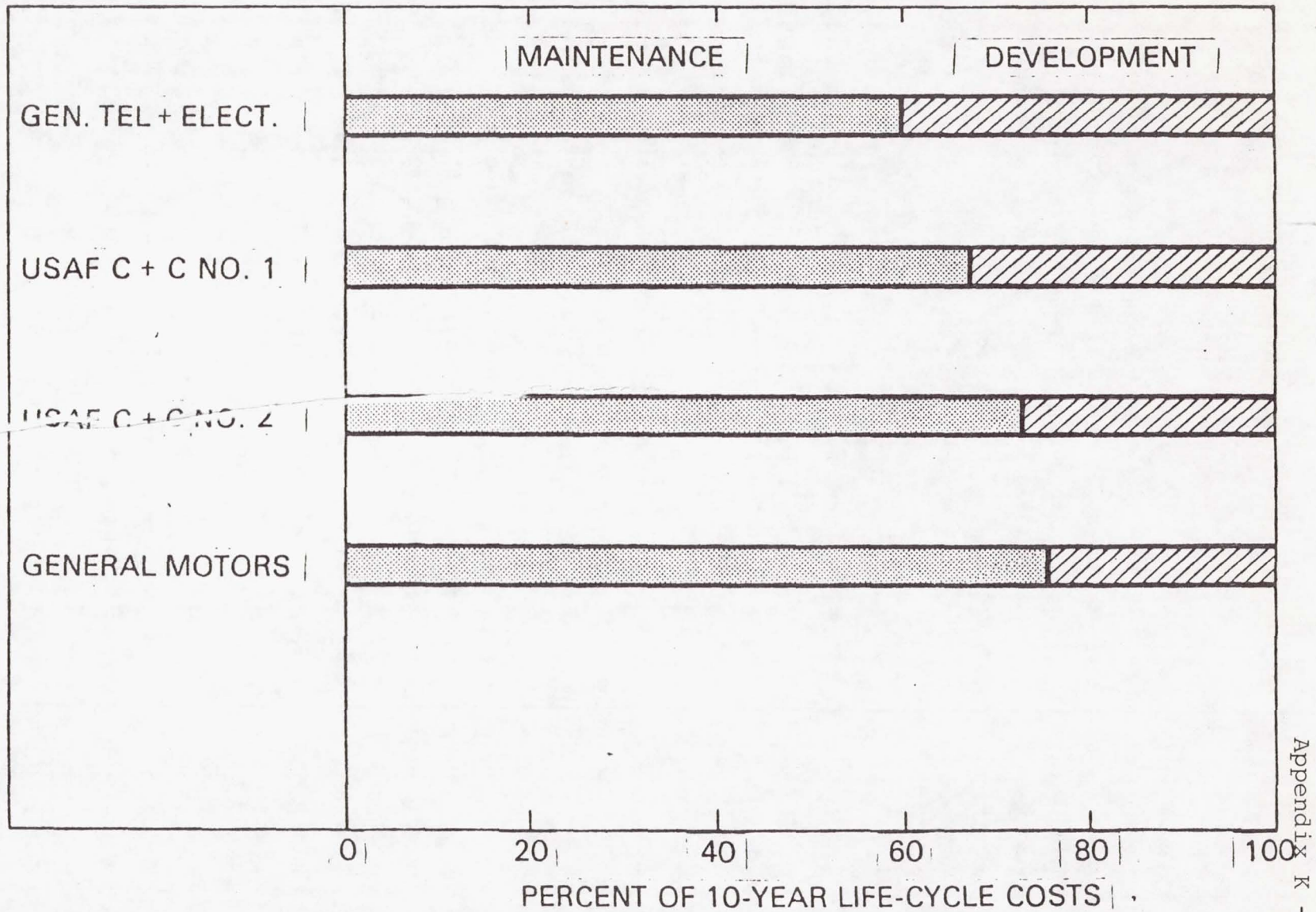


Figure K-6.

Hardware now accounts for 20% of a computer system's lifetime cost and software for the rest (Figure K-3). By 1990 the ratio may get worse, with software approaching the 90% mark (Boehm, 1979b). Data from TRW, GTE, and IBM indicate that the costs of fixing an error grows tremendously the later in the life cycle it is discovered (Figure K-4). Specifically, an error found in the coding phase costs 7 times as much to correct as an error found in the requirements phase, and 5 times more during testing than during coding (Figure K-5).

According to McGowan and McHenry (1979), perhaps nothing has motivated new software practices more than the realization of the true costs of large computer systems and where in the life cycle the costs are incurred. First, software is the dominant component in computing systems. Software now takes over 70% of the budget and it is the software rather than the hardware that determines any system response and provides essential adaptability. Second, within the software realm, maintenance costs now exceed development costs. At GM, 75% of their software effort goes for maintenance (see Figure K-6). In general, software maintenance accounts for 50-80% of the total life cycle cost of a software system. Third, the detection and correction of errors account for a major portion of system cost. MITRE Corporation estimates 50% of total life cycle costs are tied directly to errors. Fourth, requirements and design errors are by far the most crucial and costly. It is thus important to concentrate whenever possible on the requirements/design phases of software development. The cheapest way to fix a software mistake is not to make it in the first place.

Software complexity is growing exponentially while productivity has been growing linearly. Software is currently labor intensive and will probably remain so for many years. According to a Texas Instruments survey (Schindler, 1981), there are currently 200,000 software people in the United States and roughly 10,000 computer science graduates will be produced per year for the next decade. The projected demand by 1990 will be for between 1 and 2.4 million programmers. Given the current trend, it is estimated that by 1990 there will be a 10 to 1 gap in the United States between the number of people needed to produce software and the number of trained people actually available. Even if all the nation's engineering graduates (50,000 per year)

became programmers they could not fill the gap.

K.4 Software Development Techniques

Modern software management employs many techniques proven effective in the software development process. Many of these tools assist the management function, and others assist in solving the technical problem of generating a correct and proper software system. Some examples include:

- o Management planning and control of software packages such as work breakdown management and PERT/CPM.
- o Software cost estimate modeling programs.
- o PSL/PSA (see below).
- o Software design languages such as PDL and SDDL.
- o Interactive computing with text editing.
- o Document processing systems for generation and maintenance of documentation.
- o Standard mathematical libraries.
- o Structured programming languages.
- o Cross-compiler systems.
- o Programming systems where editors, compilers, and debugging aids are integrated together (e.g., UNIX).
- o Program performance analysis packages.
- o Optimization packages.

Several techniques for software development are now in operation. Most familiar is the chief programmer concept used at IBM (Mills, 1979). This is at top-down structured programming approach with one person (the chief) responsible. In general the approach has been successful, but it has been difficult to find one person with enough energy and talent to lead the team. Boehm (1979b) gives examples of several other techniques used at Softech, TRW, SDC, and SPL.

Two other important techniques in software development are structured programming and the top-down approach to system design. Dijkstra first introduced the concept of structured programming in 1968, and since then

structured coding has become an important technique used by all "leading edge" software developers. In the top-down design, the programs that control interfacing of functional modules are written and tested first, then the functional modules are added, level by level, with testing and integration accomplished during each level. Conversely, in bottom-up development the functional modules are written first but cannot be tested in the final system until the end of the development.

A large number of methods, techniques, and tools have been developed to address various aspects of software development and evolution activity. Initially these approaches focused on the coding activity, but more recently efforts have been made to cover all phases of the software life cycle from the initial system concept through testing and system modifications. Uncontrolled proliferation of techniques and tools has demonstrated the need for an integrated, systematic approach to software development environments. Some of the tools available during the software life cycle are discussed below.

Requirements -- Currently software requirements are usually expressed in English, but there are several requirements languages also in use: Structured Analysis and Design Techniques (SADT) by Softech, Problem Statement Language/Analyser (PSL/PSA) from the University of Michigan, Planning Procedures to Develop Systems (PPDS) by Hitachi, and Software Requirement Engineering Program (SREP) by TRW.

Design -- Software design is still almost completely a manual process. The top-down approach has worked well and provides a way of organizing a system which focuses early attention on the critical issues of integration and interface definition. Flow charts remain a popular aid for design representation, although they have deficiencies particularly in expressing interfaces. Hierarchical Input-Process-Output (HIPO) represents relationships between modules but is largely a manual technique. Systems Architects Apprentice (SARA) is another manual tool using modules and limited connection between them.

Coding -- The list of tools available includes compilers, editors, preprocessors, and the high level languages such as FORTRAN.

Testing -- Generally, testing is not done until after the coding has been written. In general, the high cost of testing is a result of the high cost of rewriting the code at this stage. Additionally, much of the testing activity is still a tedious, manual process which is itself highly error-prone. There are many tools available and so only a few types are mentioned below (after Boehm, 1979a).

- o **Static code analysis:** Tools include the usual compiler error diagnostics, plus data type checking, as well as set-use analysis of variables, and there are also programs to check control flow and reachability.
- o **Test Case Preparation:** There have been attempts to automate the generation of test data to make a program execute along a desired path. One drawback is that these programs generate only inputs, so the tester must still calculate the expected outputs himself.
- o **Debugging:** Traditional aids include the core dump, the trace, the snapshot, and the breakpoint. Another tool is interactive walkthrough or backtracking. Again, it is still a manual task to check whether the outputs are correct.

Maintenance and Modification -- These are still largely manual tasks, limited by the time lag between the original design and coding and the need for making changes. The team must first understand the original program, make changes to it, and then test the new version for correctness.

There has been a proliferation of tools, all claiming to help improve software productivity. Still the need for some coordinated approach for the selection and use of tools exists. Some of the problems with software tools are as follows:

- o The tools should be evaluated for their effectiveness. This is done on a small scale at the Software Engineering Laboratory at NASA/Goddard, but more needs to be done.
- o Tools are generally not portable. A tool developed on one computer may

not work on another computer.

- o Many tools, although effective, are not user-friendly. This greatly inhibits their acceptance.
- o Very few of the tools are integrated so that a combination of them can be used together.
- o The high cost of learning a new tool discourages the use of tools.
- o The environment is forced to fit the tool.
- o Often it is the tool that is driving the requirement.
- o The tool capacity is misrepresented or misunderstood.
- o Some tools are too general and try to solve too many problems.

Clearly, a useful software tool will be one that operates in a minicomputer environment, is highly interactive, is relatively independent of an operating system, does not require elaborate peripheral devices, has a relatively short learning period, is relatively inexpensive to operate, is easy to maintain and enhance, and is portable.

While much has been done in developing tools, it is generally agreed that more work is needed in the tools area and particularly in their integration into a user friendly and supportive environment for the management and development of software. Tools also need to be made more portable.

NASA should continue to test tools, both for its own use and for its contractors to use, and should provide specific incentives for its contractors either to develop new tools or to use existing ones as part of a portable, integrated package. NASA should encourage industry and universities to conduct software research, and then use these new techniques and tools as they are developed.

K.5 Conclusions

NASA is a technical agency with perhaps more computer science expertise than any other government agency. To perform its future missions it will need computer science capabilities well beyond what it has now. The field of computer science is growing rapidly and will continue to do so, doubling in volume perhaps every two to three years.

Software Guidelines and Policies

The software development effort within NASA is heterogeneous. Each Center participates in the process with different organizations, different contractors, and different standards. Although the benefits of software engineering techniques are generally recognized, there is no clear agreement on which techniques are most useful or on how they should be applied. This lack of uniformity is also reflected in the contractual structure. Most contractors have implemented their own standards for software production.

But the state-of-the-art in software engineering is encouraging. Within the last decade, software development methodology has demonstrated great improvement in reliability and reduction of maintenance costs. It seems clear that uniform application of the best current software engineering techniques would produce immediate cost reductions for NASA.

For NASA, software management is the most important component of software engineering. Modern software management methodologies are understood and employed in many software development areas within the Agency. NASA Headquarters has recognized the importance of this area and has developed some guidelines for flight projects (NMI 2410.6.), but more work clearly needs to be done. NASA should be in a position to formulate and adopt reasonable guidelines for software management techniques, software tools, interfaces between tools, documentation, portability of programs, and so forth. To achieve effective software management, these guidelines should be reinforced by the following policies:

- o Formal training for software managers.
- o Supported mechanism for software technology infusion.
- o Effort to develop NASA software environment.
- o Coordinated effort to evaluate software engineering practices.
- o Coordinated effort to advance software engineering practices.

NASA Software Database

There is a general lack of interactive, integrated, portable software

development environments within NASA. Thus, it is suggested that NASA conduct a survey of the software engineering and technology activities of Agency employees involved in the development of various tools, techniques, methods, and standards supporting the software development process. The purpose of this effort is to establish a basis for coordinating software engineering and technology activities among the NASA Centers, with the aim of eliminating wasteful duplication and of keeping abreast of new developments. Following the survey, a software specialist should be assigned to coordinate software activities within NASA. A database should also be established and made accessible to users for sharing software information resources which are constantly updated.

NASA Position on Ada

In 1975 DOD began an effort aimed at reducing the rapidly increasing expense of military software systems. Four of the original goals (Brender and Nassi, 1981) of the DOD common high-order language effort were to:

- o Address the problem of life-cycle program costs.
- o Improve program reliability.
- o Promote the development of portable software.
- o Promote the development of portable software tools.

The effort has evolved into one of software engineering's most exciting and far-reaching developments -- the Ada programming language and associated support environment.

Ada was intended principally for embedded computer applications but was also deemed suitable for general systems programming, real-time industrial applications, general applications programming, numeric computation, and for teaching good programming practice. The facilities in Ada have been formulated to provide mechanisms for modularization.

Ada incorporates and directly supports modern programming concepts of abstraction and modularization, separate compilation of program units without loss of program-wide checking, concurrency, and features for efficient systems

programming. Since Ada has widespread support of both DOD and many commercial hardware and software vendors, it could possibly affect the industry in a significant way. Thus, it is recommended that NASA initiate a plan which will prepare the Agency to take full advantage of the potential benefits of Ada in its future projects.

K.6 Recommendations

Four recommendations are given which, with future budgets uncertain, need not be fully implemented to get useful results. A small investment will yield generous improvements in the present situation.

1. **NASA should expand its efforts to test and to evaluate new software techniques.** The National Bureau of Standards maintains a Software Tools Database listing over 300 tools. Tools are often designed for a specific task and may work well in one environment but not in another. The Software Engineering Laboratory (SEL) at Goddard is a good example of a software testing facility, but more needs to be done. It might be desirable to have some type of SEL at each Center. These labs could also act as clearinghouses for software information (see Recommendation 2), and also as nodes on a NASA software network (see Recommendation 4).
2. **NASA should monitor itself, industry, and academia for new developments in software engineering.** It is important for any high-technology organization to keep abreast of new developments. While there have been many efforts in this direction by individual NASA personnel, there has been little coordination or sharing of results. The coordination might be done through the SEL, and the sharing of results could be done via some NASA-wide network.
3. **NASA should encourage new developments in software engineering.** The software realm in the United States is large (over \$20 billion annually), and NASA plays a relatively small part in it (less than \$500 million per year). NASA can participate in, but cannot hope to drive or lead, software development. Major breakthroughs are not likely to come from NASA, and its role should be to continue to encourage industries and universities to do

software research. Grants to universities should be continued and expanded if possible. NASA's software contractors have a vital interest in using modern techniques, and it is recommended that software contracts target money specifically for research and improvements in software development. Thus, a contractor could implement new techniques or improve existing ones, NASA would ultimately get better software, and contractors would get more efficient operations.

4. **NASA should plan for software exchange.** There are currently several efforts underway by government agencies to establish databases for software information. The National Bureau of Standards maintains a Software Tools Database listing over 300 tools, and the GSA maintains a Software Exchange Program for sharing software programs among government agencies. However, without appropriate incentives to encourage the submission of a program to the database and to encourage the use of the database, such a database will be ineffective. As a first step towards a NASA software database, the Agency needs to recognize that tools and programs must be portable, user friendly, and often be part of a larger integrated system before they can be shared effectively.

K.7 References

Belady, L.A., and Lehman, M.M., "Characteristics of Large Systems," in P. Wegner, ed., Research Directions in Software Engineering, MIT Press, 1979, pp. 106-138.

Boehm, Barry W., "Software Engineering," in E. Yourdan, ed., Classics in Software Engineering, Yourdan Press, 1979, pp. 325-364.

-----, "Software Engineering: R&D Trends and Defense Needs," in P. Wegner, ed., Research Directions in Software Engineering, MIT Press, 1979, pp. 44-86.

Brender, R.F. and Nassi, I.R., "What is Ada?" Computer, June 1981, pp. 17-25.

Dijkstra, E.W. and Hoare, C.A.R., Structured Programming, Academic Press, 1972.

Martin Marietta Aerospace "CASID General Requirements for Computer-Aided Specification/Integrated Development," working paper, 1981.

McGowan, C.L. and McHenry, R.C., "Software Management," in P. Wegner, ed., Research Directions in Software Engineering, MIT Press, 1979, pp. 207-253.

Mills, H.D., "Software Developments," in P. Wegner, ed., Research Directions in Software Engineering, MIT Press, 1979, pp. 87-105.

Schindler, Max, "Software," Electronic Design, January 1981, pp. 191-199.

Teichrow, D. and Hershey, E.A., "PSL/PSA," in E. Yourdan, ed., Classics in Software Engineering, Yourdan Press, 1979, pp. 389-410.

U.S. General Accounting Office, "Contracting for Computer Software Development," NTIS, PB80-105638, November 1979.

U.S. General Accounting Office, "Wider Use of Better Computer Software Technology Can Improve Management Control and Reduce Costs," FGMSD - 80-38, April 1980.

Appendix L. Decision Support Systems

A Decision Support System, or DSS, is a computer-based tool to assist executives, managers, analysts, and professionals in performing their jobs more effectively and efficiently. The function of DSS is to assist in planning, in analyzing decision alternatives, and in monitoring the progress of activities. The tool addresses nonstructured rather than structured tasks, provides support to a decision-maker rather than replacing the judgment of the DM, and focuses on effectiveness rather than efficiency in the decision process (Alter, 1980).

L.1 Elements of a Decision Support System

A computer-based DSS usually consists of four elements. A database management system (DBMS) controls the storage and access of data by the system. A query language allows users to interact with the DBMS. Applications programs permit users to operate on the data using statistical techniques or planning models. Finally, a report writer allows users to specify how their output is to be presented.

The DBMS controls the access to system data. Even if these data are distributed over a network, the DBMS can still locate and utilize them. The DBMS must service operational needs other than the DSS in most systems, so the database can be separated into two parts -- operational data and management data (Keen and Morton, 1978). Operational data relates to detailed transaction information. These data must be detailed, up-to-the minute, and accurate. Management data may be summary, historical, or current. The DBMS may segment the database into the operational data which belongs to operating organizations (such as parts of NASA Centers), and then create management data files accessible by Headquarters or other offices using a DSS. Data integrity is obtained by retaining ownership of the file and control of access, and micromanagement can be limited by controlling the data entering the management files.

The query language enables users to interact with the system to determine what they wish done. There are two major considerations. First, the language

or query system must be simple or user-friendly so that a manager can use it on his own terms. One should not need to be an expert on the system to use it; the system should adapt to the manager's terms and style, rather than vice versa. Secondly, the language should be rich enough to satisfy diverse user needs. The query language enables users to select from the database the specific entries that are desired. How this is done will depend on the organization and structure of the database.

Once a user has created a file with the desired data from the database, he can then operate on the file using a variety of applications programs. These programs might assign values from probability distributions, perform statistical analyses, perform a simulation, or perhaps add an inflation adjustment for budget/plan values. Finally, the report-writer allows the user to specify what information is to be displayed, how it is to be displayed, and on what medium it is to be presented (e.g., hardcopy printer, terminal display, or Calcomp).

The DSS operates as an executive computer system enabling the user to specify which tasks are to be accomplished. The system must be user-friendly. It should be flexible enough to allow for users with differing degrees of familiarity and knowledge to use it, from the rank novice who needs complete guidance to the very sophisticated, knowledgeable user who can use direct command language. The system should be designed to minimize user frustration for the novice. One approach is to have the user type a plain text description of what he wants and have the system conduct a dialog until it understands exactly what is intended. Another popular approach presents a user with sets of menus from which he selects the option(s) desired.

L.2 Use of a Decision Support System

A DSS may be used by a planner, project manager, supervisor or analyst of any kind. Some administrative assistants may also use them. It is designed to aid in solving unstructured problems where the analyst does not know exactly what the answer is or perhaps even the most appropriate form the question (Sprague, 1981).

The flexibility of a DSS will aid the creativity of a decision-maker or analyst. The system will stimulate thinking and aid in the development of new alternatives and options. A good system allows these new ideas to be tested through analytical models or simulation and can result in courses of action which would not otherwise have been discovered (much less considered).

L.3 NASA Implementation of Decision Support System

NASA is beginning to introduce some DSS technology into the organization. It is important that this be done in an informed, coordinated manner if the Agency is to benefit from the capabilities of these systems.

It is an article of faith by proponents of decision support systems that they must evolve from the desire and need of the users (Carlson, 1977). Systems that are imposed without consideration of the responses of the users are just not used. The problem is how to have such systems evolve from employee demand throughout NASA, but evolve in such a way that hierarchical access to the systems is preserved.

Several of the Centers and Headquarters offices are developing DSS technologies for their own use. It would be beneficial if Headquarters could have access to some of the information and capabilities at the level of individual Centers. This could be accomplished in several ways. For instance, NASA could promulgate a standard which would provide for consistency so that all DSS could be accessed by Headquarters or other Centers. Alternatively, specific translators could be provided as interfaces between specific Center or Headquarters DSS's. A third option, one already used by PRMS, provides for the Centers to construct files from their systems which can then be accessed by Headquarters. Whichever is selected, NASA could make a conscious selection and pursue it vigorously as soon as possible.

L.4 References

Alter, S.L., Decision Support Systems: Current Practices and Continuing Challenges, Addison-Wesley Publishing Co., 1980.

Carlson, Eric D., "Decision Support System: Personal Computing Services for Managers," Management Review, January 1977, Reprinted in D.H. Sanders, S.J. Birkin, eds., Computer and Management in a Changing Society, McGraw-Hill Book Co., 1980, pp. 249-255.

De. P. and Sen, A., "Logical Database Design in Decision," Journal of Systems Management, May 1981, pp. 28-33.

Keen, P.G.W. and Morton, M.S.S., Decision Support System: An Organizational Perspective, Addison-Wesley Publishing Co., 1978.

Sprague, Ralph H., Jr., "Decision Support Systems: ICP INTERFACE, Administrative & Accounting, Summer 1981, pp. 23-26.

Appendix M. Assessment of Computer and Communication Systems in the Private Sector

Fortune magazine's 500 largest industrial corporations, 50 largest life-insurance companies, and 50 largest commercial banking companies were the base from which the private sector representatives were chosen. The following four specific questions were asked of all of the companies surveyed in regard to their computer and communications systems modernization programs (if any):

- (1) What proved to be the most critical part of your program?
- (2) What was thought to be important that didn't turn out to be so important?
- (3) What unexpected problems did you encounter?
- (4) How was the program received by the organization?

The results and the assessment of each company are summarized below.

M.1 Industrial Companies

All seven industrial firms surveyed have overseas operations and have experienced some difficulty with their overseas networks. Most companies use ATT lines but S.B.S. is beginning to replace some of the ATT lines because of reduced cost. There results of the assessment are as follows.

Automotive

- o Transmission -- lease lines and direct dial
- o Protocol -- uses its own requires vendors to use it, except railroad
- o Terminals -- interactive
- o Electronic mail -- yes
- o Decentralized

Petroleum

- o Transmission -- lease lines and direct dial; digital and voice (10% data, 90% voice)

- o Protocol -- computer supplier's
- o Terminals -- interactive
- o Electronic mail -- yes
- o Decentralized
- o Questions: (1) Reliability of components
(2) Lead time installation and security
(3) Manual intervention and international communications
(4) Substantial resistance at beginning of program

Electronic and Appliance

- o Transmission -- lease lines and direct dial
- o Protocol -- computer supplier's
- o Terminals -- interactive
- o Electronic mail -- beginning
- o Decentralized
- o Growth -- 30% to 35% annually in volume
- o Questions: (1) Lead time
(2) Security
(3) System reliability 97% when 99+% required
(4) Acceptance good due to proper planning
- o Additional comments -- This company is on its third generation of its computer and network system. All three phases have been fully planned and implemented.

Metal

- o Transmission -- lease and direct dial, changing to S.B.S.
- o Protocol -- computer supplier's
- o Terminals -- interactive
- o Electronic mail -- yes
- o Centralized

Conglomerate

- o Transmission -- lease line and direct dial

- o Protocol -- computer supplier's
- o Terminals -- 75% are interactive
- o Electronic mail -- yes
- o Decentralized

Aircraft

- o Transmission -- owns private microwave and Earth Station
- o Protocol -- computer supplier's
- o Terminals -- interactive
- o Electron mail -- yes
- o Decentralized (five major centers, link net)
- o Growth -- 30% to 35% annually in program
- o Questions: (1) Switching function
(2) International, trouble reporting and tracking
(3) Loading of system and memory problems
(4) Very good reception

Electronic and Appliance

- o Transmission -- private microwave at head office; lease lines and direct dial for rest of network.
- o Protocol -- computer supplier's
- o Terminals -- 25% are interactive
- o Electronic mail -- yes
- o Decentralized
- o Questions: (1) Implementation, adjustment problems
(2) Security
(3) People, learning how to maximize system use
(4) Should have no problem with new system
- o Additional comments -- This company is putting in new state-of-the-art network systems. It is all-digital which ties together all headquarters and plants within a 30-mile radius. Access to this network will be possible with all other operations in the system.

M.2 Commercial-Banking Companies

Three representative commercial-banking companies were assessed for their experience with internal communications and computer networks. The results are as follows.

New York

- o Transmission -- lease lines and direct dial
- o Protocol -- computer supplier's
- o Terminals -- interactive
- o Electronic mail -- some
- o Decentralized

Chicago

- o Transmission -- lease lines, direct dial, & hard wired (data)
- o Protocol -- computer supplier's
- o Terminals -- interactive
- o Electronic mail -- yes
- o Decentralized
- o Questions: (1) People
 - (2) Graphic modeling, automatic calendar
 - (3) Response time
 - (4) Very good reception
- o Additional comments -- The Vice-President of Systems interviewed saw no executive work stations in the future, just a chair, couch and telephone. He believes he will have a small terminal in his briefcase and will come to the office only a couple of times a week, doing most of his work at home. Another concept to enhance acceptance of the system (and of electronic mail) would be for an IBM sales team to sell personal systems to the employees.

San Francisco

- o Transmission -- lease lines, direct dial, and satellite
- o Protocol -- computer supplier's
- o Terminals -- partly interactive
- o Electronic mail -- no
- o Centralized, but going to regional structure
- o Growth -- 20% annually in program
- o Questions: (1) Availability and integrity
 - (2) None
 - (3) Mean time of recovery
 - (4) Educational process for personnel

M.3 Life Insurance Companies

Three representative life insurance companies were assessed to determine their commitment to the use of state-of-the-art communications and data processing systems. The firms are all doing business primarily in the United States, with a little business in Canada. The assessment summary follows.

Newark

- o Transmission -- lease and direct dial
- o Protocol -- computer supplier's
- o Terminals -- interactive
- o Electronic mail -- 45% are interactive
- o Decentralized
- o Growth -- 30% to 35% annually in program
- o Questions: (1) Reliable software
 - (2) User's needs
 - (3) User's needs
 - (4) Changes were accepted

New York

- o Transmission -- lease lines and direct dial

- o Protocol -- computer supplier's
- o Terminals -- interactive
- o Electronic mail -- yes
- o Decentralized
- o Questions: (1) Management problem analysis
 - (2) Phone service
 - (3) People
 - (4) Very good acceptance

Hartford

- o Transmission -- lease lines and direct dial, ATT (data), S.B.S. (voice)
- o Protocol -- computer supplier's
- o Terminals -- interactive
- o Electronic mail -- no
- o Highly centralized
- o Growth -- 25% annually in program
- o Questions: (1) Availability to user and response time
 - (2) Privacy
 - (3) Loading of system
 - (4) Wide acceptance

Appendix N. The NASA Manager Survey

The study of NASA management and executives reported in this Appendix was designed and executed to explore the receptivity of key Agency personnel to increased computerization and their perception of the role of computers in NASA's operations. It was decided that a structured telephone interview, conducted with a reasonably large sample of management/executive personnel, would be the most appropriate given the time and various other constraints. Besides receptivity to computers, other issues -- many of which are not directly addressed in this Appendix -- are reflected in the construction of the "agree/disagree" queries on the questionnaire (particularly items #29-37). The findings in some of these categories are reported in this text. The complete survey instrument is reproduced at the end of the Appendix, and the characteristics of the sample of managers who took part in the survey are summarized in Figure N-1.

The sample included NASA Headquarters, because of the concentration of management and executive personnel at that location, and representatives from a geographical distribution of Centers. Two of the Centers are heavily involved in aeronautics work while the other two concentrate on space systems work. This pattern of selection was used to reduce potential bias.

The sample, although not random in a technical sense, was taken without regard to name (except one) or position. Of the original 56 names selected for interview, 63% (35) usable interviews were actually completed; 21% (12) of the interviews were not completed because of refusals, retirements or vacations, and the remaining 16% (9) were inaccessible for various reasons.

The 35 interviewees cannot be properly characterized as representative of NASA management, and therefore the conclusions are not generalized. Although the 35 cannot be considered a random sample, the range of years of service with NASA and the distribution of educational backgrounds do show that a wide variety of individuals were captured in the sample. This gives some confidence that the sampling procedure did reduce some of the bias. Generally, those interviewed have served with NASA for a long time, typically 16-20 years. Figure N-2 shows the distribution of their Agency service time.

Figure N-1. NASA Manager Survey Sample

<u>Location</u>	<u>Number Selected</u>
NASA H.Q.	16 (28.6%)
Ames	10 (17.8%)
Langley	10 (17.8%)
Goddard	10 (17.8%)
Marshall	10 (17.8%)
	--
Total	56

Figure N-2. NASA Management: Years of Service

<u>Service</u>	<u>Number</u>
1-5 years	2
6-10 years	3
11-15 years	10
16-20 years	13
21-25 years	5
26-30 years	2

Figure N-3. NASA Management: Educational Levels

<u>Level</u>	<u>Number</u>
Doctorate	6
Masters	14
Bachelors	13
High School	2

The highest formal education levels attained by those interviewed, shown in Figure N-3, ranged from High School diplomas at Ph.D.'s.

In addition to their formal degrees, almost all interviewees had participated in job-related training within the last five years; most indicated it was considered "part of the job description." All had extensive management experience and in most cases it had been with NASA exclusively; only a very few of the interviewees had recent management experience from outside the Agency.

N.1 Findings of the Survey

The questionnaire covered several areas. The first concerned perceptions of NASA's mission or purpose, any changes anticipated in that mission and any changes that ought to occur in NASA's mission. The overwhelming majority of those interviewed indicated that they believed NASA's mission to be research and development in the field of aerospace, and the technology transfer related to that research and development (see Figure N-4).

The managers did not anticipate future changes in NASA's mission nor did they want changes in that mission. The few that did say that a mission change was in the offing indicated that the changes would be brought about by "environmental" influences. The sources of change mentioned were: (1) Militarization of NASA's functions, (2) the economy, and (3) addressing more practical concerns. Computing was not mentioned as a source or change in the mission of the Agency. In addition, those who have suggested that NASA's function is data collection and dissemination should not look to these managers for support. The "service agency" concept generated little enthusiasm, as can be seen in the data at Figure N-5. These responses are only suggestive of the opposition to the "service agency" concept -- additional comments were often strongly critical of the notion.

Although computers were not seen as a major source of change in NASA, they were recognized as a salient feature of Agency management. The data showed that, individually, most of the managers spent 5% or less of their time in direct computer usage. However, they are dependent on computers for critical

data, reports, and process control. Computers are necessary for NASA management to get their work done, and all who are interviewed considered them a vital tool. Many also felt that computers could and should be used for additional management support.

When asked to evaluate a series of specific computer-based systems as effective or not effective for a person doing their kind of work, the managers generally evaluated such systems as effective. Figure N-6 shows the results of their evaluations. A comparison of the Index values in Figure N-4 indicates that "effective" evaluations far outweighed "not effective" evaluations.

Of the 34 usable interviews on this question, 94% felt that word processing was effective -- indeed, many volunteered "very effective." Computerized databases received an effective evaluation by 91%, while computerized management information systems received an effective evaluation by 71% of the interviewees. The most disturbing result was the magnitude of the "no evaluation" response in some categories, which indicates a lack of familiarity or understanding of specific kinds of systems. For example, 74% gave no evaluation of computerized decision support systems, and 53% gave none for electronic mail. Lack of familiarity may have a more substantial impact on how management approaches computerized systems than purely negative evaluations.

The majority of NASA managers interviewed were not dissatisfied with their internal and external information exchange systems and in almost every case no recommendations for change were made. However, when asked about better alternatives for generating or accessing computerized data, expressions of dissatisfaction emerged. Interactive access to data, rather than hard copy, was often urged as a superior alternative to present procedures of collecting large piles of hard copy computer printout and extracting and reprinting data from that output. If NASA were to move toward more interactive data access, it would constitute a major restructuring of computer usage that would also require more extensive interfacing of management with computers.

It is also interesting to note that nontechnical managers were

Figure N-4. NASA As A Research and Development Agency

NASA should continue to be a research and development agency with the primary purpose of fostering and demonstrating aerospace capabilities.*

Agree	100%	(35)
Neutral	0%	(0)
Disagree	0%	(0)

* Question #29, see attached interview schedule.

Figure N-5. NASA As A Data Service Agency

NASA should become a service agency with the primary purpose of gathering and disseminating data using aerospace technology.*

Agree	11.4%	(4)
Neutral	5.7%	(2)
Disagree	82.8%	(29)

* Question #30, see attached interview schedule.

Figure N-6. Evaluation of Computer Based Systems

	Effective	Not Effective	N.E.*
Word Processing	32	0	2
Comp Networks	20	1	13
Electronic Mail	14	2	18
Comp Tele Conf	18	1	15
Comp Data Bases	31	1	2
Comp M.I.S.	24	4	6
Comp D.S.S.	6	3	25
	--	--	--
INDEX**	4.26	.2	2.38

* N.E. - No Evaluation, not familiar with system or concept.

** INDEX - Total response in column/total interviews (34)

proportionately less satisfied with their systems than technical managers. Possibly this is because technical managers can justify computer acquisition as part of projects while nontechnical managers have greater difficulty in justifying computer acquisitions.

Based on the above data and other information gleaned during the survey, it is impossible to characterize the managers who were interviewed as unreceptive to new or expanded computer usage. It also appears, however, that careful preparation must precede any major changes which are to be made. Managers must be educated as to the benefits of newer systems and be assured that "past mistakes" will not be repeated. The past mistakes were characterized as "...letting computer systems dictate management functions rather than managers dictating computer systems." In addition, there seems to be some feeling that promises or claims for computerized systems did not match actual performance (see Figure N-7). This strongly argues for the need to test systems or to be able to quickly reject systems that prove burdensome. It may even imply that lease or lease-purchase of equipment and software is necessary in order to avoid "sunk cost" arguments that militate against change.

The educational need may be addressed by providing access to short courses or seminars that deal with the functions and operations of decision support systems, computer networks, or electronic mail systems. Since such educational experiences appear to be a normal part of the activities of those interviewed, it suggests that a mechanism is already available for this purpose.

Finally, it should be noted that the proposed creation of an organizational component of NASA that would concern itself with advancing the state-of-the-art in computers did have support from those interviewed. However, that support was often qualified by placing it in the context of the aerospace mission. Statements which accompanied opposition to the development of such an organization in NASA argued that "...industry can do that better than we can" (see Figure N-8).

Figure N-7. Experience With Computer Based Systems

Experience with computer based systems suggests that their benefits were frequently not as great as was anticipated.*

Agree	37.1%	(13)
Neutral	17.1%	(6)
Disagree	45.7%	(16)

* Question #31, see attached interview schedule.

Figure N-8. Computer Organization Support

NASA should develop an organization which has the capability of advancing the state-of-the-art in computer systems and technology.*

Agree	48.6%	(17)
Neutral	20.0%	(7)
Disagree	31.4%	(11)

* Question #33, see attached interview schedule.

N.2 Conclusion

The initial problem was to explore the receptivity of NASA managers to change that may be brought by computerization. Based on the people interviewed, there is little reason to believe that management would not be receptive, although the door is not wide open. It is clear that for these people a studied introduction would be necessary to establish and to maintain their confidence. But the potential for acceptance of a phased introduction of new computer systems to meet NASA's future needs is very high.

Appendix O. OPEN and ST: Two Future NASA Projects

Two major projects planned by NASA for the 1980's are OPEN (Origin of Plasmas in the Earth's Neighborhood) and ST (Space Telescope). Both are in the planning stages and thus it may still be possible to incorporate new and innovative ideas into them.

O.1 The OPEN Project

The existence of a magnetic field around Earth has been known for centuries, and its shape was supposed to be similar to that of the magnetic field due to a bar magnet. The correlation between the appearance of sunspots and other solar activity and the terrestrial phenomenon of aurora and interference in radio communication has also been known for a number of decades. In 1957 a pioneering discovery was made by James Van Allen using radiation counters aboard Explorer-1: Earth is surrounded by belts of high-energy particles trapped in Earth's natural magnetic field high above the atmosphere. Further experiments in the 1960's and 1970's led scientists to ask such questions as: "What is geospace?", "How is it formed?", and "How does it behave?"

It is now known that different regions of the geospace are populated by entirely different types of plasmas. High-energy electrons fill the radiation belts encircling Earth in the equatorial regions. Another component of hot ionized gas is found in the comet-like tail of Earth's magnetic field. On the day side, a funnel-shaped region is formed in the magnetosphere near the poles (the polar cusps) that allows high-energy particles to stream down to Earth's surface. However, this picture of the geospace is still a sketch, so a multi-spacecraft program is necessary to collect data on simultaneous events throughout geospace as a whole.

Two plasma source regions (the solar wind and the ionosphere) and two storage reservoirs (the extended geomagnetic tail and the inner plasma sheet/particle trapping regions) are known. These four regions represent the four pieces of the modern geospace puzzle, interconnected by a network of transport processes that give rise to the overall structure of the geospace.

QUESTIONNAIRE

NASA MANAGER SURVEY

NAME: _____

POSITION: _____

1. How long have you been with NASA? _____ yrs.
2. (if less than ten years) What was your previous position before joining NASA?
3. How long have you been in your present position? _____ yrs/mo.
4. (if less than two years) What was your previous position with NASA?
5. What is your highest academic degree? _____
6. When did you complete that degree? _____
7. Have you taken any additional college or university work since you completed your _____? When? _____
8. Have you taken any noncollege job-related training since you completed your _____? When? _____
9. In how many professional associations do you hold membership? _____;
10. Which of these associations do you consider more important to you?
11. Have you ever taught at a college or university? no _____; yes _____; part-time _____; full-time _____.
12. Briefly describe what you believe to be NASA's mission or purpose.
13. Do you believe that there will be a major change in NASA's basic mission or purpose in the next 10-15 years? yes _____; no _____.

14. (if yes) What do you believe will be the most important reasons/causes for the change(s)?
15. In the future, do you believe that NASA's mission/purpose ought to be different than you described? yes____; no____. Please explain.
16. Briefly describe your work activities and responsibilities.
17. Briefly describe the number and kinds of people that work with you in your area of responsibility.
18. On the average, what percentage of your time do you use computers or computer controlled equipment/processes in your work?_____%
19. Do you prepare reports or proposals that use computer generated data or append computer output as part of the document? no____; yes____; approximate \$/yr._____.
20. (if yes) Do you feel that there are better alternative ways in which computerized data can be prepared for your use?

21. Do you receive reports from other NASA offices that use computer generated data or append computer output as part of the reports?
no____; yes____; approximate \$/yr._____.
22. (if yes) Do you feel that there are better alternative ways in which such data can be given to you?
23. In general, do you believe that you have effective means of exchanging information with others inside NASA? yes____; no____; why? why not?
24. In general, do you believe that you have effective means of exchanging information with others outside NASA? yes____; no____; why? why not?
25. On the average, what % of the time do others in your area of responsibility spend in the use of computers or computer controlled equipment or processes?_____%
26. Do you know of any NASA projects or pilot projects that are designed to introduce or implement new computer usages at your center: NASA H.Q. yes____; no____.
27. (if yes) Can you name or describe them?

28. I am going to read a list of items and I would like for you to evaluate each item as effective or ineffective for a person doing work like you do. If you are unfamiliar with the items, you may indicate no evaluation.

<u>ITEMS</u>	<u>EFFECTIVE</u>	<u>INEFFECTIVE</u>	<u>NO EVALUATION</u>
1. word processing	_____	_____	_____
2. computer networks	_____	_____	_____
3. electronic mail	_____	_____	_____
4. computerized teleconferencing	_____	_____	_____
5. computerized data bases	_____	_____	_____
6. computerized M.I.S.	_____	_____	_____
7. computerized decision support systems	_____	_____	_____

The following is a series of statements in which you are asked to express agreement or disagreement. You may also indicate whether you are strongly in agreement or strongly in disagreement. If you have no feelings concerning the statement, you may indicate that you are neutral.

29. NASA should continue to be a research and development agency with the primary purpose of fostering and demonstrating aerospace capabilities.

SA-----A-----N-----D-----SD

30. NASA should become a service agency with the primary purpose of gathering and disseminating data using aerospace technology.

SA-----A-----N-----D-----SD

31. Experience with computer based systems suggest that their benefits were frequently not as great as was anticipated.

SA-----A-----N-----D-----SD

32. The cost of systems support staff tends to outweigh the savings that may be brought through office automation.

SA-----A-----N-----D-----SD

33. NASA should develop an organization which has the capability of advancing the state-of-the-art in computer systems and technology.

SA-----A-----N-----D-----SD

34. Computer programmers are generally unsympathetic to the problems faced by management.

SA-----A-----N-----D-----SD

35. The implementation of computerized management support systems throughout NASA would increase H.Q.'s ability to monitor center activities.

SA-----A-----N-----D-----SD

36. The overall productivity of NASA management would be greatly enhanced by an infusion of advanced computer technologies that are presently available to management in the private sector.

SA-----A-----N-----D-----SD

37. Computer programmers tend to be overconfident and promise more than they can deliver.

SA-----A-----N-----D-----SD

By placing properly instrumented spacecraft in each of the four regions, scientists hope to understand the storage and release of energy, the entry and transfer of plasmas in the different regions, and how they change with time.

The four spacecraft designed for this purpose are:

- (1) **Interplanetary Space Laboratory (ISL)** which will measure the incoming solar wind, magnetic fields, and particles. It will be placed at a point 1.5 million kilometers sunward from Earth where the gravitational field of Earth balances that of the Sun. On reaching the stable libration point, it will be placed in a small circular "halo" orbit about this location. It will then accompany Earth around the Sun and serve as an early warning system for the appearance of solar magnetic storms.
- (2) **Polar Plasma Laboratory (PPL)** which will measure solar wind entry, ionosphere output and the deposition of energy into the neutral atmosphere at high altitudes. PPL will be placed in a highly eccentric polar orbit with a maximum altitude of between 25,000 and 90,000 kilometers.
- (3) **Equatorial Magnetosphere Laboratory (EML)** which will measure solar wind entry at the sunward lobe of the magnetosphere, and record the transport and storage of plasma in the equatorial ring current and near-Earth plasma sheet. EML will be placed in an elliptical orbit in the equatorial plane with the maximum height ranging from 6,000 to 70,000 kilometers.
- (4) **Geomagnetic Tail Laboratory (GTL)** which will measure solar wind entry and acceleration, transport, and storage of plasma in the geomagnetic tail. The GTL will utilize novel navigational methods that involve an occasional kick from the gravitational field of the Moon and keep it in an orbit around Earth that varies from Earth-Moon distance of 380,000 kilometers to points as far as 1.5 million kilometers.

Two important factors will help in making OPEN a successful project.

First, the earlier findings will help in realistic planning and sound strategy for this mission. Second, the technology needed for these measurements has now become available. The OPEN project will be a central element in the overall solar-terrestrial exploration planned for the 1980's.

0.2 The ST Project

The second major space science undertaking is the Space Telescope (ST) program. The project is under active development and the ST will be launched in the mid-1980's. It will be the first large optical telescope (2.4-meter aperture) which will be placed outside of Earth's atmosphere. It will have several advantages over other ground-based instruments. It will be able to make observations not only in the visible region, but also in the infrared and ultraviolet regions. Being outside of Earth's atmosphere it will not be affected by haze, clouds, and atmospheric turbulence, thus it will be able to see objects 50-100 fainter than those visible using Earthbound telescopes. ST can resolve astronomical phenomena within the 0.001-1 second time-resolution range. The telescope will be diffraction-limited; being outside of Earth's atmosphere, it will be able to provide sharper images than any other telescope.

In relationship to other telescopes used in previous space missions, the ST will be different in at least four significant ways: (1) It will be a full optical telescope (previous telescopes have been used only in the wavelength regions where the Earth's atmosphere is opaque); (2) ST is much larger than previous instruments, of which the largest had an aperture of only 0.6 meters; (3) ST is expected to have a 20-year lifespan, with Shuttle maintenance capability if necessary; and (4) ST will be primarily a guest-observer facility, whereas previous instruments were operated exclusively for the Principal Investigators (PIs).

Because the ST will have a long operational life and will be primarily a guest-observer facility, NASA has followed the recommendation of the Horning Committee and has set up Space Telescope Science Institute (STSI) which will be operated by a consortium of 14 universities. The Johns Hopkins University has been chosen as the site for STSI, and is already operating. The STSI will be under contract to NASA, and its responsibilities will include soliciting

and screening observing proposals, funding of guest observers, planning and scheduling of observations, execution of science quick-look functions during observations, science data processing and calibration, science data archiving, cataloguing and data retrieval, and delivery of data products to users.

The Science Institute will eventually have a staff of about 200 people, of which about 40 will be astronomers. The staff will not have privileged access to the observation time but will have to compete with other proposals. Initially about 200 proposals will be accepted annually by the Institute.

The establishment of the STSI will make the data handling, processing, storage and retrieval more systematic and efficient. The basic data from the ST will consist of engineering and science data. The data will first be sent to the Tracking and Data Relay Satellites, then to the White Sands Receiving Station, then via NASCOM to Goddard's Data Capture Facility, and then finally received by STSI. At the Institute, the data will be kept on magnetic tape, with data analysis performed using magnetic disk media. Data archiving will be done on 6250 bpi magnetic tape. New methods for data storage, such as the laser disk, will also be explored and implemented as they become available commercially. However, at present the laser disk technology is not sufficiently well-developed to make it superior to the magnetic media. The data will be retrieved by a database management system.

It is anticipated that there will not be any full-fledged data preprocessing facilities on board the ST, but there will be local intelligence for targeting purposes. This involves getting a target acquisition image, on-board processing of the image to determine the location of the target, and then repointing the telescope so that the image is positioned at the center of the aperture of the telescope. There will also be some other preprocessing features on the ST such as averaging of successive detector readouts (with bad readout rejection), synchronous averaging of pulsar signals, exposure meter control (in which the cumulative exposure must reach a preset level before the exposure is terminated and the next one is begun), and error correction encoding (which inserts check bits into the data stream so data may be recovered in the event of telemetry errors).

Quick look imaging data will be available immediately at the STSI and high-quality images within 24 hours. This will facilitate planning and updating of the observational strategy. The data will be considered privileged information for a period of one year, then will be available publically.

The Space Institute is planning to develop transportable software. This will have a dual advantage. First, the guest observer will be able to do data analysis and reduction at his home institution. Second, the same software will be usable for future generations of computers. The data will be distributed in a star network to all users -- that is, they will all be connected to the STSI but not to one another. The data will also be distributed in a standard format and thus general database routines are being developed for this purpose. At present, the data handling facilities are being developed by Computer Science Corporation as a subcontractor.

The data will be calibrated to a standard level by the STSI. However, the calibration needed for a particular observation, and the necessary algorithms, will be the responsibility of the guest observer. Flexible image transport system, which is already in use at Kitt Peak National Observatory, will be used for imaging. It is anticipated that about two reels of 2400 foot, 6250 bpi magnetic tapes will be generated each day.

The establishment of the STSI is an important step in overcoming some of the forthcoming basic data handling problems of NASA. The fundamental issues are being addressed right now and adequate remedies are being proposed well before the problems get out of hand.

Appendix P. The High Cost of Computer Science Expertise

A major conclusion of the 1981 Summer Study is that NASA needs to develop in-house expertise in computer science in addition to the expertise presently scattered throughout the various Centers. This expertise is essential whether or not the Agency decides to undertake a research and development activity in computer science. The field of computer science is so volatile that it is impossible to stay abreast of developments throughout the discipline unless Agency personnel remain closely associated with scientists in industry and academia who are relatively working in research and development.

At present computer science is undergoing significant quantitative advances every two years and significant qualitative changes at a much faster rate than the twelve-year cycle attributed to most sciences. A NASA computer scientist who spends five years working with fixed equipment and software problems is competent to work with that equipment and software, but he will have difficulty stating realistic requirements for new systems, will have trouble knowing which vendor's promises are trustworthy and which are not, and may not be fully aware of new developments which are just over the horizon unless he has been able to maintain a wide range of contacts external to NASA.

In order to acquire and retain the kind of people who are willing and able to stay abreast of the field, NASA must compete effectively with other government agencies, academia, and industry. The Agency must exploit its advantages -- an exciting image, extraordinary projects, and extensive arrays of equipment. It must also overcome its handicaps -- low salaries, a sluggish bureaucracy, lack of a career ladder for computer scientists, a service or support perspective on the role of computer scientists, and obsolescence of much of its equipment. By way of comparison, most other government agencies have many of the same problems. However, they lack the exciting image and do not have programs as scientifically interesting as those with which NASA deals. Academia has many of the same problems as the Agency but offers people extreme freedom in choosing their activities. Industry offers much higher salaries and a much more up-to-date work environment, but also offers considerably less freedom of activity than academia.

This Appendix is an attempt to bring the reader up-to-date on the state of affairs in employment at the doctoral level for computer science. Of course the Agency needs personnel beneath the doctoral level, but the leadership will likely come from those having doctoral or equivalent training and experience.

The number in Figure P-1 are taken from an annual survey of doctorates granted in computer science, conducted by Orrin Taulbee and S.D. Conte for the Computer Science Board (Taulbee and Conte, 1979). The 1977 and 1978 estimates were obtained from Arden (1980). The 1980 estimate of 265 Ph.D's granted is taken from Denning (1981a) and its source is unknown. The alternative 1980 estimate of 341 is taken from Taulbee and Conte (1980). The extreme optimism present in the 1977 and 1978 figures suggests the lower 1980 estimate is probably closer to reality.

Figures P-2 and P-3 show the employment of computer science doctoral graduates. Our sources do not contain figures for 1978 or 1980. Notice the decline in the total number of Ph.D's entering the academic environment. Also the number of computer science Ph.D's entering government is miniscule. Note that despite an annual average of 70 to 80 new doctoral graduates entering Ph.D-granting departments, the net increase is only 20-30 per year. This suggests that about 50 computer science faculty have been leaving academia annually. Dr. George D. Dodd of General Motors Research Laboratory recently noted that:

In 1979 200 Ph.D computer scientists were graduated from American universities and colleges, down from 256 in 1975. Of these 200, more than 100 are being absorbed into industry and governmental positions, leaving the remainder for university positions. Surveys and studies conducted by the Computer Science Board and others have concluded there exists a demand for 600 Ph.D computer scientists per year to meet educational needs and 1300 per year to meet total U.S. needs. We are indeed in a crisis nationwide. (Estrin, 1980)

These figures are incomplete. Nonetheless, they strongly suggest that NASA

Figure P-1. Supply of PhD Graduates

<u>Year</u>	<u>CS PHDs granted</u>	<u>Estimates</u>
1973	208	
1974	203	
1975	256	
1976	246	
1977	208	310
1978	214	299
1979	248	
1980	265	341

Figure P-2. Employment of Computer Science PhD graduates

<u>Year</u>	<u>PhD granting Depts</u>	<u>Non-PhD granting Depts</u>	<u>Government</u>	<u>Industry</u>
1975	84	53	12	69
1976	76	51	9	94
1977	66	48	12	75
1978				
1979	73	36	8	122

Figure P-3. Number of Faculty Holding PhD

<u>Year</u>	<u>Number of Faculty Holding PhD</u>	<u>Change</u>
1974	787	
1975	805	18
1976	773*	-32
1977	790	17
1978	825	35
1979	837	12

* The structure of the survey changed.

must expect difficulty in recruiting and much more rapid turnover among personnel in CS&T than in other fields.

It is widely believed that salaries are the driving force behind these figures (Denning, 1981a). In 1980, average industrial-setting salaries were \$20,000 (B.S. degree), \$24,000 (M.S. degree), and \$32,000 (Ph.D degree). Even for ordinary programmers, salaries have skyrocketed: The national average salary for the DP field rose 12.8% in 1980 (Lusa and Winkler, 1981). Examples of these salaries in 1981 were: Applications programmer trainee, \$14,800; Senior systems programmer, \$23,800; Systems Analysis Manager, \$33,900. By comparison, new Ph.D's were offered an average 9-month salary of \$22,000 for academic positions.

The computer science community and the government have recognized this problem for several years. The Feldman Report (1979), resulting from a November 1978 NSF-sponsored workshop, argues that there is a crisis in experimental computer science. The Snowbird Report (Denning, 1981b), resulting from the July 1980 biennial meeting of computer science department chairmen, states:

There is a severe manpower shortage in the computing field. It is most acute at the Ph.D level: The supply of new Ph.D's is about 20% of the demand. The crisis has been precipitated by an explosive growth of the computing field with no matching growth of university budgets in computer science. Unless the trend reverses, the country will soon lose its lead in computer technology because enough computer experts cannot be trained and because the basic research to ensure a continuing supply of new concepts for the long-term future cannot be conducted.

In October 1980 the Office of Science and Technology in the executive Office of the President released a report entitled Science and Engineering Education for the 1980s and Beyond, which was requested from NSF and ED (Education Department) by President Carter. This report singled out the computing profession as one with special, pressing problems; it foresaw shortages of computer professionals persisting into the 1990s. Finally, in

March 1981, a consensus statement issued by the NSF-supported "Panel in Computing and Higher Education" stated:

There is a major crisis in training computer scientists. There is a lack of support to graduate students, which leads to a shortage of faculty. New capital is also required for the equipment to support these needs. Already there is a crisis in computer science -- the faculty members needed to teach the growing numbers of students are preferring to enter industry in order to have access to up-to-date research facilities. This results in the twofold problem of not enough graduates with advanced degrees and not enough faculty to provide computing education. (Estrin, 1981)

NASA will have to make special efforts to provide competitive salaries and other incentives in order to continue to attract talented and qualified people. Furthermore, if NASA should take the role of lead government agency in driving the development of computer science in this country, it should recognize the critical importance of increasing the production of computer science trained people at all levels.

References

Arden, B.W., What Can Be Automated?, COSERS, MIT Press, 1980.

Denning, P.J., "Eating Our Seed Corn," Communications of the ACM, Vol. 24, June 1981a, pp. 341-343.

-----, "A Discipline in Crisis," Communication of the ACM, Vol. 24, June 1981b, pp. 370-374.

Estrin, G., "Responding to a National Crisis in Computer Science," UCLA Computer Science Department Quarterly, Vol. 8, Summer 1980, pp. 1-6.

-----, "The Crisis in Computer Science Has Not Evaporated," UCLA Computer Science Department Quarterly, Vol. 9, Winter 1981, pp. 1-6.

Feldman, J.A. et al., eds., "Rejuvenating Experimental Computer Science," Communications of the ACM, Vol. 22, September 1979.

Lusa, J.M. and Winkler, R.S., eds., "DP Salaries Looking Up!" Infosystems, Vol. 28, June 1981, pp. 38-50.

Taulbee, O.E. and Conte, S.D., "Production and Employment of Ph.D's in Computer Science -- 1977 and 1978," Communications of the ACM, Vol. 22, February 1979, pp. 75-75.

-----, "Production and Employment of Ph.D's in Computer Science," unpublished draft, January 1980.

Appendix Q. An Inside View of CS&T in NASA: Problems and Potentials

During the 1981 Summer Study a large number of Agency personnel came into contact with the faculty fellow participants, and numerous frank discussions of the problems and potentials of computer science and technology in NASA were thus initiated. Though sometimes controversial, this Appendix presents the candid concerns and issues of importance to the NASA participants themselves during the course of the Study, expressed in their own words.

Q.1 Research Productivity

The productivity of a research scientist or engineer engaged in either experimental or theoretical work can be significantly enhanced by the aid of computers. At present, a typical researcher currently uses a computer to mathematically model some physical phenomenon or to display and analyze experimental data. Movies and color video displays are often used to provide physical insight into the processes. Expanded use of computers can create an environment more conducive to productivity. A researcher, for example, needs information from variety of sources -- such as books, journals, reports, experiments, computational results, and colleagues -- which could be obtained through a computer network.

Computers may be used by researchers for analytical studies and for information storage and retrieval of data. Areas for improving computer facilities for research include:

- o An intelligent computer terminal which can provide easy access to computer facilities, libraries and bibliographic searches, tools for software development and graphical display, electronic mail capability for communication with colleagues, text processing for report preparation and editing, tracking of administrative information, ready access to a summary of key results of other research, a calendar of events and action items, and a log of research findings.
- o Improved computer graphics tools.
- o Robust mathematical libraries.
- o Special-purpose high-level languages (e.g., PDE).

Just as a typewriter and a telephone are essential tools for the secretary, so an intelligent computer terminal should be available in the office of every NASA research scientist and engineer.

Q.2 Institutional and Scientific Data Management

NASA has and will face an ever-increasing information problem. This problem is (at the very least) a threat to efficient operation of the organization, and to the successful completion of its missions. We can look at the problem from two perspectives -- an institutional information perspective or a scientific information perspective.

From the institutional perspective, existing procedures are too dependent on hardcopy, do not provide sufficient or timely access, are too fragmented, lack commonality among similar components, require hand transfer of information, and store redundant data. Among these are administrative procedures such as payroll, personnel, procurement, financial and resource management, property inventory, and the like; interoffice communication; information retrieval; the manual processes involved in report generation; and project management tools for smaller projects which frequently include haphazard determination of items such as critical path. The net result of the above is an organization whose affinity for the "old way" is stifling the administrative and management functions.

From the scientific perspective, at least two concerns are paramount: (a) Information exchange in a timely manner, and (b) the collection, storage, manipulation, retrieval, and distribution of scientific data. These data have many origins (e.g., analytical studies, experimental data, satellite transmissions), diverse characteristics, and varying degrees of usage. Any effective study must deal with large volumes, high bandwidths, ability to browse or scan, means of insuring integrity and security, identification and categorization of data, the multidisciplinary nature of users, the manipulation of large blocks of data in the face of commercially available systems tuned to managing many identical small block data sets, the file level incompatibility of data management tools and user programs, remote access and transmission of data, data presentation, and the geographically distributed

nature of both data and users.

Q.3 Flight-Crucial Systems

Digital flight control systems are typical of flight-crucial systems in which a generic software fault (i.e., a system failure not handled correctly by software) can be catastrophic, resulting in loss of life or loss of substantial resources. Systems of this kind must be developed in a more methodical and systematic manner than conventional digital systems. Guidelines for developing such systems, as well as up-to-date experiences with real life problems; need to be shared among the various groups in this business (i.e., several NASA Centers, the control system houses, and the airframers). Many of these guidelines must be applied early -- in the specifications. For example, the requirement to have good visibility into the system must be called out, or the visibility won't be there when it is needed during checkout. During the development of the system, there are frequently geographically-dispersed organizations involved where carefully controlled transfer of information is necessary, such as control system constants or simulation databases. Guidelines for carrying out verification and validation of flight software need to be shared among the various NASA Centers such that confidence is built up within all parties of a particular project and the most systematic approach is taken to ferret out software bugs early, before they become responsible for delays along the critical path. Typical questions are:

- (1) How many of the hardware elements need to be included in the simulation?
- (2) How much testing is required at each stage of development?
- (3) How much retesting when a change is made just before first flight?
- (4) How many of the failure cases need to be tested in closed-loop simulation?

Q.4 End-to-End System Engineering

End-to-End System Engineering (E2SE) deals with the process and discipline by which large, multi-user, information-intensive systems are initiated, specified, implemented, and operated. It deals with the definition, application, and control of the means for responsively dealing with user requirements, expectations, and uncertainties; sponsor's goals and constraints; technology limitations and possibilities; and human capabilities

and frailties -- all toward the realization of stated goals. Such goals include implementation, transfer to operations, and maintenance of the particular system. EESE may be viewed as a finite set of "black boxes" (people, machines, and data) which receive and process (or store) input data (or events) and produce output data (or actions) which, hopefully, contribute toward the intermediate and final system products, either directly (by implementation) or indirectly (as in management and administration).

Each such "black box" is thus a transformation and transmission medium that must be dealing with inaccurate (noisy) input, missing information, and faulty, untimely control (management). It may be overloaded with more input than it can handle, or be called upon to process beyond its capability. It may respond to given requirements or constraints erroneously, or to unstated, assumed requirements or constraints needlessly. These black boxes are connected via a complicated, seldom well-defined network of relationships mirroring system design, project organization, and administrative organization. Data, project controls, and products flow through this mesh in a rather stochastic fashion that generally trends toward the classical "top-down" engineering cycle -- that is, from requirements capture to analysis, to functional design and resource allocation. (Maintenance may be viewed as the same kind of EESE process, but with a large inherited baseline, and perhaps fewer -- and perhaps also less well-defined -- pieces than the original development).

The EESE process must cope with many problems, chief of which are:

- o System (task) complexity.
- o Organizational complexity and burden.
- o Requirements complexity, errors, omissions, and conflicts.
- o Communication overhead (meetings, reviews, paper flow volume and timeliness, and so forth.
- o Interface mismatches between EESE elements.
- o Process irregularities (revolting development, requirements dynamics).
- o Technology transfer, portability, reuse.
- o New technology adoption, adaption, utilization; not-invented-here syndrome.

- o Fiscal and schedule constraints, overruns.
- o Near-term expediency vs. long-term payoffs.
- o Inadequate tools, aids, and models for EESE.
- o Poorly trained personnel, failure to keep current.
- o Inertia in everything: Sponsors, users, managers, and staff.
- o Institutional burdens (e.g., procurement practices).
- o Software quality and reliability.

Q.5 Software Development Productivity

As hardware technologies improve at a spectacular rate, the problems of software development are changing. Computer systems will become widely available with greater processing power and more storage capability, with multiple microprocessors handling functions which were once software functions, with fault-tolerance in hardware systems providing more reliability, with continued evolution and breakthrough certain in hardware. In general, these systems will provide more productivity and reliability for the hardware dollar.

This hardware revolution is putting the pressure on software development to become cost-effective and reliable. In a NASA flight project the software development may be the driving cost. This problem may be particularly evident in NASA-peculiar software, while outside of NASA a pool of portable, usable software will be available. The problem NASA faces is the development of capabilities in the following areas to deal with software development in the future for flight projects:

- o Requirements specifications.
- o Portability.
- o Software tools for specification, construction, testing, analysis, management, documentation, and maintenance.
- o Reliability and quality assurance.

Q.6 Shuttle Planning and Operations

The era of Shuttle operations has a principal objective of supporting

approximately one vehicle launch per week with a resource of four vehicles. Functional facets involve mission planning and scheduling, vehicle preparations, flight crew preparations and operations, and vehicle recovery and status assessment. Characteristically, the development and implementation of these functions involves:

- o The interaction of factions which have geographic, organizational, and discipline-oriented differences.
- o Vehicle hardware/software modifications and functional validation to accommodate mission and performance requirements.
- o Vehicle, ground-support equipment, payload integration, and checkout preparatory to launch.
- o Real-time flight data assessment for vehicle configuration management and contingency reactions to off-nominal conditions.
- o Flight crew and flight controller training/upgrading.

Scheduling responsiveness dictates the need for increased automation and/or more efficient techniques. Budgetary limitations create the need for increased levels of autonomous operations and greater productivity. A key issue to all of the above includes the information management of complex and voluminous documents involving definition and documentation of information or requirements, distribution for interdisciplinary coordination, revision to reflect responses, and final approval and distribution for usage. Information criticality dictates review by affected organizations, and usage criticality necessitates formalized management scrutiny and final configuration approval.

The issue of autonomous and automatic operations must consider safety of manned flight. The probability of correct operations must approach 100%. Intelligence is needed to resolve conflicts between redundant actions or reactions, and the ability to reconcile transient off-nominal conditions (soft failures) is essential.

Launch Operations in the Shuttle Era

STS launch operations comprise vehicles, cargo, and ground support activities from orbiter landing to launch. These activities include orbiter

deservicing, vehicle stacking, cargo integration, rollout, countdown, and launch. All components of the STS must be tested and monitored during these procedures. As the program matures, ground turnaround times must decrease from several months to a few weeks. It is required that launch operations be performed on multiple orbiters, in turnaround time of weeks, with fewer personnel on the launch team. This problem is complicated by the fact that each mission will be unique. Cargo operations are tailored to the specific payload, and it is estimated that for the first few years of operation, 20% of the vehicle procedures will be modified (and validated) for each flight. In addition to the direct operations, all support activities will be affected by the high launch rate. Scheduling and logistics, for example, must become more efficient. If these requirements are to be fulfilled, it seems necessary to develop and apply innovative techniques to automate launch procedures.

Shuttle Documentation

Typically, manned space vehicle programs employ a comprehensive documentation system to manage the program and to maintain configuration control of the vehicle system during the phases of development -- design definition, development, certification, and mission operations. The documentation system can be highly interactive with some elements becoming quite dynamic during specific program phases.

Looking at the Space Shuttle system, it is noted that there are four major program elements involved -- the orbiter vehicle, the external tank, the solid rocket boosters, and the launch processing system. The documentation system involved is basically as follows. At the top is a set of program documents referred to as level II. These define overall Shuttle and program requirements, Shuttle operations and performance, and criteria on requirements for software systems. Next are Level III documents which control a specific element. The orbiter vehicle end item (OVEI) specification is typical.

The OVEI specification is levied on Rockwell (the orbiter manufacturer) and is the basis for design, development, and construction of the orbiter. It established (1) the basis for individual subsystem design requirements (e.g., avionics hardware and software, mechanical structures, environmental control,

propulsion, power), (2) the environmental certification requirements, and (3) the reliability and quality criteria. Based on the OVEI specifications, Rockwell develops individual hardware specifications and contacts with suppliers to build and test the items.

Software is unique. There are four programmable devices on the orbiter and each is managed differently. They are: (1) The five general-purpose computers (GPC), (2) the four multifunction CRT displays, (3) the three main engine controllers, and (4) the two Pulse Code Modulation (PCM) instrumentation units. JSC (IBM) specifies, designs, and verifies the software operating system for the primary GPC's. Rockwell defines the applications requirements which are coded and implemented by JSC (IBM). The display system and PCM unit software are Rockwell responsibilities. The engine software is a MSFC responsibility. Its design integration with the orbiter system is via an interface control document (ICD) with Rockwell, who identifies the orbiter GPC functional support requirements.

The Shuttle development phase activity requires verification testing which is a key to the decision to launch. Successive verification levels and activities include:

- o Black box qualification.
- o Software verification.
- o Avionics hardware/software integration.
- o Integrated avionics and flight control.
- o Manufactured vehicle checkout prior to delivery.
- o Crew, vehicle, and ground controller integration.
- o Pre-launch checkout and countdown demonstration tests.

Preparatory to all flights is the activity of mission planning and mission operations definition which are driven or influenced by flight test requirements, and the composite of vehicle and subsystem performance capabilities and software design. Outputs from mission planning typically consist of (1) trajectories and flight dynamics definition, (2) mission events, (3) logistics, and (4) contingency planning. Mission operations planning outputs are typically (1) crew procedures, (2) consumables

management, (3) vehicle configuration management, and (4) malfunction procedures. The mission planning and operations output becomes the guiding document to accomplish flight crew and ground controller training, and are utilized for the actual mission.

The above dialogue is intended to identify major activities which are formally recorded by the documentation system, which is characterized by its need for wide geographic distribution, review by numerous contractor and NASA organizations, and involvement by varying engineering and management disciplines. In the case of new missions, the program management, mission operation, or payload requirements which differ from the baseline must be properly documented, resulting in a series of activities dependent upon the scope of the change and the verification level required.

The manual involvement for ground servicing and flight operations concerns (1) the requirement for continuous flight monitoring of vehicle and crew performance and for control of mission activities by complex ground elements, and (2) the magnitude of vehicle recovery and turnaround operations by the launch processing system program element. For the initial research and development phase on new programs, high manual involvement is appropriate. For an operations phase, it is widely acknowledged that more advanced computer techniques must be employed to reduce the on-line, real-time mission operation support by orders of magnitude. Likewise, ground turnaround must employ computer techniques to streamline operations. Highly reliable automated decision-making methods, along with automated configuration management and automated scheduling of activities in nominal or off-nominal conditions typify the technology needs.

Mission Operations

Mission operations, in this context, refers to the support activities required to fly a spacecraft and to deliver a data product to the user community. It is a continuous process at GSFC for approximately 30 spacecraft, and can be characterized by a recurring sequence of activities:

- o Provide planning data to investigators, e.g., spacecraft trajectory,

- status, day/night;
- o Investigators request spacecraft events;
- o Spacecraft sequence is generated;
- o Sequence is merged with other sequences to form operational sequence for tracking and communication system (global);
- o Control Center generates command sequence which is uplinked to the spacecraft; tracking and telemetry data are taken (scientific and housekeeping);
- o Ancillary data (attitude and orbit) are completed and merged with scientific data;
- o Scientific data are provided to investigators; and
- o The cycle is repeated.

This is a highly simplified portrayal of a large support capability which involves well over a thousand people in a continuing process of system operation, maintenance, and enhancement. It involves many separate and distinct subsystems -- spacecraft control centers, network control center, orbit computations, attitude computations, command sequence management, image processing, and so forth. All of these are managed separately and involve distinct computer systems, many of which are very old (10-15 years). It must support a geographically distributed user population, and interfaces with them primarily through voluminous hardcopy (e.g., planning data) and magnetic tape (e.g., scientific data). Planning and scheduling is a substantial portion of the activity, done currently through largely manual means with some computer aids. Interface control among the subsystems is a growing problem (most are manual or tape). Operations are split between crude interactive and batch -- no graphics. Response time to users is long (weeks); operations are labor-intensive; software is largely custom. A continuous support capability must be maintained 24 hours a day, every day of the week. There is a large dependence on contracted manpower with high turnover rates, and there are many external interfaces to other Centers and agencies.

Appendix R. Report of the Goal-Setting Workshop for the
1981 Computer Science Summer Study

Conducted at the
Xerox International Center for Training and Management Development
Leesburg, Virginia
31 May 1981 -- 5 June 1981

Preliminaries

FINAL REPORT

TABLE OF CONTENTS

Title Page	1
Table of Contents	2
List of Participants	3
1. Introduction	5
2. Management Tools	7
2.1 Immediate Opportunities	7
2.2 Longer-Term Opportunities	7
3. Networking	9
4. Recommendations for a Computer Sciences and Technology Activity	12
4.1 Definitions	12
4.2 Payoffs and Opportunities	13
4.3 Computer Science Technologies	14
4.3.1 Software Engineering	14
4.3.2 Space Data Management and Distribution	14
4.3.3 System Architecture	15
4.3.4 Interactive Systems and Graphics	15
4.3.5 Modeling and Numerical Computation	16
4.3.6 Office Procedures	16
4.3.7 Theory of Computation	17
4.3.8 Artificial Intelligence (AI) and Automation	17
5. Possible First Steps	18
6. References	20

LIST OF PARTICIPANTS

Daniel S. Slotnick (Workshop Chairman), Room 283 D.C.L., University of Illinois at Champaign-Urbana, Champaign, IL 61801

Harlow Freitag (Workshop Vice-Chairman), IBM Corporation, 1000 Winchester Avenue, White Plains, NY 10604

Management Tools Team

Richard Harden (Co-Leader), 7732 Bridle Park Lane, McLean, VA 22102

George Ainsworth-Land (Co-Leader), 230 North Street, Buffalo, NY 14201

Harvey Cragon, Texas Instruments, P.O. Box 225012, Dallas, TX 75265

James P. Hogan, 435 South Shepard Street, #2, Sonora, CA 95370

Networking Team

James Burrows (Leader), Institute for Computer Science and Technology, National Bureau of Standards, Washington, DC 20234

Virgil D. Gligor, Department of Computer Science, University of Maryland, College Park, MD 20742

Donald C. Wilt, Hughes Aircraft Company, Space and Communications Group, Bldg. S50, P.O. Box 92919, Los Angeles, CA 90009

Computer Science and Technology Team

Raymond T. Yeh (Leader), Department of Computer Science, University of Maryland, College Park, MD 20742

James S. Albus, Industrial Systems Division, National Bureau of Standards, Washington, DC 20234

Robert A. Freitas Jr. (manuscript editor), Space Initiative/XRI, 100 Buckingham Drive, Suite #253A, Santa Clara, CA 95051

Carl Helmer, North American Technology, Inc., Strand Building, Suite 23, 174
Concord Street, Peterborough, NH 03458

Chitoor V. Ramamoorthy, Division of Computer Science, University of California
at Berkeley, Berkeley, CA 94720

J. T. Schwartz, Computer Science Department, Courant Institute of Mathematical
Sciences, 251 Mercer Street, New York, NY

Summer Study Planning Group

University of Maryland

Raymond T. Yeh (Co-Director, 1981 Summer Study)

Virgil D. Gligor

NASA

Paul Schneck (Co-Director, 1981 Summer Study), Code 500, Goddard Space Flight
Center, Greenbelt, MD 20771

Rodger A. Cliff, Code 402, NASA Goddard Space Flight Center, Greenbelt,
MD 20771

Ronald L. Larsen, Code RTE-6, NASA Headquarters, Washington, DC 20546

Preliminaries

1. Introduction

The workshop was convened to produce a set of goals for the ten-week 1981 Computer Science Summer Study which immediately follows. It has available, from the beginning, materials reporting two previous studies (Sagan, 1980; Long and Healy, 1980). Although agreeing with several of the conclusions and recommendations of these reports, the present workshop maintained a position which is, perhaps, at some variance: That NASA exists to fulfill its role in aeronautics and space and not primarily to advance the technologies it uses to accomplish its missions. At the same time Workshop members noted that the prevalent NASA view of computers as off-the-shelf tools has substantially inhibited the creative and productive use of such machines throughout the Agency. In particular, Agency management must be made more aware that a system can exhibit the many benefits (and pitfalls) of newness as much by reason of its software as its hardware.

We suggest the following broad study goals:

- A. **Identify** those NASA operations where computer-based information management techniques and systems would, in the view of NASA management, fulfill the most critical set of currently unsatisfied needs. To achieve this goal a methodology is suggested which will immediately permit a broad and deep interchange between NASA personnel and Summer Study participants. The ultimate result will be the identification of relations between the array of needs to be satisfied and the technological constituents of the means to be used.
- B. **Develop** a plan for the incorporation of computer networks into those NASA operations which could most benefit from the resulting heightened levels of resource sharing. Such a plan should comprehend not only the immediate cost and performance benefits of better hardware, software, and data utilization but also the less-easily measurable benefits accruing from the concomitant establishment of a single technical community out of geographically and administratively distant professionals.

C. **Formulate** the research and service charter for a discipline-oriented (as opposed to mission-oriented) NASA Computer Science and Technology (CS&T) Organization. It is our intention that this Organization be reasonably focused on engineering approaches and engineering solutions to hardware, software and application problems within the scope of NASA activities. This is in contrast to the theoretical and abstract areas that are now and ought to remain the concern of university computer science departments.

2. Management Tools

The rapid advance of Computer Science and Technology during the past twenty years, coupled with NASA's intense and successful focus on its institutional mission, offer an opportunity to capture these advances and to exploit them for NASA's managers. NASA missions culminate in delivery of technical data to users, whereas the NASA management role begins with receipt of administrative and project data for use in planning, decision making, control and evaluation. A systems view of managerial information needs and styles will result in the design of a supporting system that frees NASA managers of many current constraints and makes them increasingly effective, more productive, and better able to carry out the Agency's mission according to plan.

2.1 Immediate Opportunities

Because of the wide-range of managerial responsibilities and styles it is not appropriate to attempt to create a comprehensive stereotype. Instead, a sample of 25 key NASA managers can be used to construct five profiles which are "typical" in terms of responsibilities, decisions required, and information needs (both personal and organizational). Each manager interviewed can characterize the need for improvements and their worth as well as cost.

Integrating the real and perceived needs of NASA management with recent advances in computer technology such as networking, databases, office automation, communications and graphics can lead to a system design that satisfies those needs. A second sequence of individual meetings are important at this time to ensure each manager recognizes that his or her stated information needs are being addressed by the program plan.

2.2 Longer-Term Opportunities

NASA has been cited frequently as a role model for management in other organizations. The introduction of advanced CS&T in support of management will enhance NASA's ability to continue in its management leadership role and will provide a basis for "exporting" Agency experience and approaches to other

agencies and national organizations.

3. Networking

The principal technical conclusion of the Workshop study group is that NASA needs to develop a single, general purpose computer and data network. While specialized, fixed-purpose nets do exist within the Agency, its geographically decentralized character clearly implies that the addition of an Agency-wide network will reduce costs and provide many qualitative and quantitative benefits. In addition, such a control network will amplify the uses of specialized and local nets within NASA, helping Project Centers to retain their local character while interfacing with all other services available.

Some benefits of developing such a single net include:

- (1) The ability to interface and coordinate the activities of NASA's users from a technical viewpoint and NASA's centers from a managerial viewpoint. This would permit new modes of interaction as well as decreased costs to NASA.
- (2) The ability to share resources. Among the resources to be shared are data, software, and specialized equipment unique to one local site. Networking makes all individual facilities available to everyone on the net.
- (3) A special case of resource sharing -- the sharing of the work product of people who access the net. A network allows the development of local "centers of competence" in specific disciplines. These centers can concentrate the personnel resource to achieve a "critical mass" effect, while making work and expertise available to all (subject to the control of the owner of the data).
- (4) Allowing easier access to existing non-NASA networks so as to make available to NASA the resources of the entire technical community.
- (5) Displacing travel costs and saving the time of key NASA technical and managerial personnel.

- (6) Making project data such as technical status and schedules accessible on a more timely and more accurate basis within the Agency.
- (7) Facilitating an electronic mail system.
- (8) Enhancing the credibility of NASA as an organization developing and exploiting advanced computer technology. In addition, it will create a more attractive image for the recruitment of new personnel.

In general, such a central network will allow NASA to maintain its Project Center and Principal Investigator orientations, while conferring many of the advantages of close interaction among these units and individuals. The technical route to the development of a network can be selected from among the many already in existence doing highly useful work, or new avenues can be explored if this is clearly preferable and necessary. It must be recognized that the netting of heterogeneous systems from different manufacturers and sources will complicate the development of a technical solution. We believe that the proposed CS&T Organization, mentioned elsewhere in this report, could play a key role in making that decision.

The benefits of netting are not free. Usually there must be major changes in the way a networking organization views and manages the information flow. One key question is network usage. In the absence of a network, none is perceived to be needed. However, its very existence suggests opportunities for effective service heretofore unanticipated. It is essential that networking usage begin modestly and grow at a natural pace. Unrealistic expectations will lead to disillusionment on the part of management. But networking represents an area where intensive usage pressure develops on the part of users, once they can experience its benefits and the additional functions it offers.

Another issue which must be addressed is network management. Alternatives include management by a committee consisting of key execution personnel from each of the local centers, or by a single organization responsible for network development and usage. While management by committee is in principle feasible, experience with real, working nets suggests that the choice might

lean towards a more centralized approach.

4. Recommendations for a Computer Science and Technology Activity

The scope of computer science is defined in this section by illustrating the problems, activities, and exciting advances now on the horizon. The difference between the study of computers and the applications of computers must be communicated effectively to NASA management, so this distinction has been highlighted. Also, a list of significant potential payoffs is provided to demonstrate the benefits to the agency if CS&T is fully embraced and utilized. These payoffs are categorized into three classes:

- (1) Improvements that can be achieved now, using CS&T techniques.
- (2) CS&T that would enable NASA to fulfill its mission in ways which cannot be done without that technology.
- (3) Areas in which NASA has a unique stake that could also contribute to national priorities in productivity improvement and automation.

A greatly abbreviated list of CS technologies potentially important to the Agency is provided, suggesting recommendations to NASA regarding possible first steps in moving towards the acquisition of an advanced in-house CS&T capability. A number of constraints are addressed (which must be overcome to achieve CS&T goals) to emphasize the desired pragmatic character of the recommended program.

4.1 Definitions

The 10-week Summer Study must define the scope of Computer Science and Technology. To communicate more effectively with NASA middle- and upper-management, this definition should highlight the distinction between computer science and computer applications.

The term "computer science" is generally used to describe that set of activities and disciplines which include system development as well as hardware and software design and construction. In the context of NASA activities this means aspects of the field that are really state-of-the-art

computer technology as distinct from the abstract and theoretical views which would be more appropriate to a University "computer science" department. We are concerned here with an engineering approach and engineering solutions to those segments of NASA's mission for which computer and system technology can be of benefit.

4.2 Payoffs and Opportunities

What can a strong, new computer technology and service program do for NASA? From a near-term point of view, there are specific practical effects which may result from the creation of an informed core group of NASA people involved in CS&T work. From a longer-term point of view, Computer Science and Technology issues are a part of every projected Agency goal and may be regarded both as a level-raiser and as a source of expertise.

A permanent non-projected-oriented CS&T program could be consulted for examination of operating assumptions and procedures used in computer areas of NASA projects. For instance, the software verification procedures used in the Shuttle program failed to detect a problem with initialization and synchronization of the spacecraft flight computers. This minor crisis might have been averted if the CS&T discipline had been represented in the STS flight software verification group.

Also in the short-term, a coherent CS&T program should be charged with education by demonstration. This activity should be treated as a model for the use of computers in everyday agency practices including word processing, telecommunications, forms management and project management so that savings by using CS&T can be clearly shown.

As a agency charged with the long-term goal of space utilization, NASA also needs a research program in CS&T. The scope of all advanced future missions will change with the infusion of the results of such a program. In fact, to achieve some of these future goals, very advanced CS&T is mandatory.

Finally, the unique motivations for NASA to acquire advanced CS&T in space automation and machine intelligence places the Agency in a unique position to

contribute substantially to national priorities in productivity improvement and general automation. This may come about if NASA simply makes available its own technology to industry.

4.3 Computer Science Technologies

There are many Computer Science technologies that could provide high leverage in both short- and long-term NASA activities. We have assembled a representative set of important examples as a conceptual aid for the reader, but without any pretensions of completeness. A number of items are important to the basic Agency mission and have been classified into eight broad groups including software engineering, space data management and distribution, system architecture, interactive systems and graphics, modeling and numerical computation, office procedures, theory of computation, and artificial intelligence.

4.3.1 Software Engineering

Software systems designed from user requirements are essential for proper operation. Proper management and quality control in software development processes are well-recognized as essential in reducing development and life cycle costs, as well as ensuring correct programs. Software development must be pursued within NASA with the recognition that software cannot indefinitely prolong the useful lifetime of ancient hardware and aging computer facilities. Subject areas of note include:

- o Software productivity and management
- o Formal verification
- o Operating systems
- o Software tools
- o Software quality assessment
- o Distributed software

4.3.2 Space Data Management and Distribution

A flight mission begins, both chronologically and operationally, when a

sensor on a spacecraft transfers data to the NASA ground system. It is completed when data are accessed by and disseminated to users. In recent years data rates and complexity of use have increased by orders of magnitude. Two issues critical to the successful utilization of space data are: (1) The evolution of the 1:1 relationship between user and data to a many:many relationship; and (2) the evolution from manual indexing and storage of data (by printed catalog and computer tapes) to on-line catalogues and databases. Particularly relevant areas of research are:

- o Database management/dissemination
- o Data networks
- o Cataloging
- o Data structure
- o Data access

4.3.3 System Architecture

Many NASA missions depend on high performance systems controlling real-time critical processes. Advances in system architecture, particularly those made possible by VLSI, could provide superior performance together with the high availability required of these systems. Issues to be addressed include:

- o Advanced-sensor data processing
- o Microprocessor-based networks
- o Project-specific advanced system architecture exploiting VLSI

4.3.4 Interactive Systems and Graphics

Interactive systems and graphics help designers achieve proper design conceptualizations and facilitate good design decisions while vastly improving user productivity. Interactive analysis facilities will assist in rapidly and completely validating the correctness of design and its responsiveness to requirements. Relevant areas include:

- o Computer-aided design/manufacturing/testing (CAD/CAM/CAT)

- o Graphics for simulation and evaluation
- o Man-machine interactive display systems

4.3.5 Modeling and Numerical Computation

Science involves the creation, testing, and refinement of models representing the real world. Previously expressed analytically only in explicit form, physical phenomena can now be implicitly modeled due to the advent of advanced computer technology. Previously insignificant numerical effects may become significant and require better understanding.

In engineering, models are used to simulate well-understood physical systems for purposes of speed, safety, and convenience. Much of NASA's modeling, by contrast, is used in an exploratory fashion as an aid in unlocking the secrets of the physical interactions. Often there is no opportunity to explore the process directly -- as in stellar evolution. NASA must be a leader in the numerical and computational techniques of modeling, such as:

- o Numerical analysis
- o Modeling and simulation techniques
- o Numerical mathematics

4.3.6 Office Procedures

The technologies listed below have proven helpful in automating certain repetitive data-driven and office tasks. Management information systems permit more effective management and cost control. Database techniques are essential in managing vast quantities of data, in analysis, and in effective utilization of data to produce information.

- o Office database management/dissemination
- o Office automation including small computers, word processing, and related simple software aids
- o Electronic mail
- o Management information systems

4.3.7 Theory of Computation

NASA's operations as they are presently conceived have a fundamental dependency upon algorithmic solutions, from relatively straightforward monitoring of spacecraft status to highly complex information extraction processes. As the trend to more highly automated systems and mission sophistication continue to grow, algorithmic dependency will likewise increase and the fundamental underlying processes become more complex.

Computational theory provides the foundation for developing efficient, correct algorithmic solutions to complex problems and, therefore, should be considered a significant technological element of CS&T within NASA. Study areas may include:

- o Logic and formal systems
- o Computational complexity
- o Efficiency of algorithms
- o Numerical analysis

4.3.8 Artificial Intelligence (AI) and Automation

For purposes of convenience we include computer-based automation in this grouping. It must at once be pointed out, however, that NASA's interest in automation commences from the lowest level of linear (non-feedback) sequence automation to efforts in robotics traditionally associated with artificial intelligence. Planned unmanned probes will continue to give NASA a unique stake in automation. Spinoffs are likely to involve other national priorities. NASA interests in AI include the following:

- o Automation systems
- o Robotics
- o Knowledge representation
- o "Expert" systems
- o Theorem proving
- o Image processing

a continuing commitment to maintain. The appointment of an outside review group should be considered to evaluate NASA CS&T needs and plans, and to provide a formal interface to the Computer Science and Technology community.

In the brief time available to the participants in the Goal-Setting Workshop, the group has attempted to outline some of the potentially fruitful areas in which NASA could reap significant benefits from CS&T and to identify some of the problems which should be anticipated. We sincerely hope our insights prove to be of some assistance to those who will be exploring this area in depth during the summer. CS&T is an exciting area, one in which we believe NASA can become an affirmative national force.

6. References

Long, James E. and Healy, Timothy J., Co-Directors, Advanced Automation for Space Missions: Technical Summary, Report of the 1980 NASA/ASEE Summer Study on the Feasibility of Using Machine Intelligence in Space Applications, University of Santa Clara, Santa Clara, California, 15 September 1980.

Sagan, C., Chairman, Machine Intelligence and Robotics: Report of the NASA Study Group, Final Report, March 1980.